Computing Modular Data for Drinfeld Centers of Pointed Fusion Categories

Angus Gruen

A thesis submitted in fulfillment of the degree of
Bachelor of Philosophy (Honours) - Science at
The Mathematical Sciences Institute
Australian National University

October 2017

Typeset in Palatino by TEX and LATEX 2$_\varepsilon$.

Except where otherwise indicated, this thesis is my own original work.

Angus Gruen
26 October 2017

Acknowledgements

It's been a great year.

First and foremost I would like to thank my supervisor Scott Morrison for all his help over the last year. There is no way that this project would have come anywhere close to completion without the countless hours he has spent helping me get both the theory down and the code base working.

Jean, Damon, Guy, Matthew, and my parents also deserve many thanks for proofreading my thesis and providing feedback; especially considering that, for many of them, I think this thesis read a bit like arcane magic.

I thank my housemates Damon, Jean, Vienna, and Alex for putting up with my mathematical rambling over the past 8 months and helping to keep me sane. On a similar note I thank my family for all their support and in particular my father, David, for encouraging me to pursue mathematics for as long I can remember. I would also like to give a special thanks to Katrina, without whom my life would be far more dull.

Finally, I doubt I would have gotten anything done without the work of Larry Page and Sergey Brin who wrote that which allows information to be found on the web.

Abstract

A theoretical background is developed to explain in detail the link between the modular tensor category $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$ and the representation category of a quasitriangular quasi Hopf algebra $D^{\omega}\,G$. Using this link, a classification of the simple objects in $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$ and formulas for the modular data of $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$ are carefully derived. Then, code is written in GAP to produce the modular data of $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$, given ω and G. This is used to create a database of modular data for the Drinfeld doubles of pointed fusion categories with dimension less than 47. This database as well as GAP code accompanying it can be found at `https://tqft.net/web/research/students/AngusGruen`. For a basic example of how this database might be used, we briefly analyse patterns in the ranks of $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$ as $|G|$ varies and produce lower bounds for the number of Morita equivalence classes of pointed fusion categories of a given dimension less than 47. For dimensions below 32, these lower bounds agree with the lower bounds published by Mignard and Schauenburg in [1]. At dimension 32 we improve upon the published lower bound and for dimensions 33 through 47 we present the first set of lower bounds on the number of Morita equivalence classes of pointed fusion categories at each dimension.

Contents

Chapter 1

Introduction

Fusion categories are an important area of study due both to their frequent occurrence in category theory as well as their applications in physics. An important current area of study in low-energy physics is the study of a phenomenon called "topological phases of matter". These topological phases are potentially key components in technological advancements in areas such as superconductors and quantum computing. One approach to studying topological phases of matter is through modelling them with topological quantum field theories (TQFTs). The cobordism hypothesis [2] states that TQFTs are classified by higher categorical data and in particular the work of Douglas–Schommer-Pries–Snyder [3] identifies fully extended TQFTs in 2+1 dimensions[1] with fusion categories.

Pointed fusion categories are a particular class of fusion categories possessing several useful properties. These categories are classified by pairs (G, ω) of a finite group G and a 3-cocycle ω on G. In addition, the Drinfeld centre of a pointed fusion category turns out to be equivalent to a representation category of a Hopf algebra known as the twisted quantum double of the group G. This correspondence allows for categorical problems to be translated over into algebraic problems which can then be solved by algebraic means.

This thesis is broadly split into three parts. Chapter 2 gives an overview of the background category and algebra theory. In particular the category theory section builds up to the definition of a modular category and gives an overview of the Drinfeld centre construction. The background algebra focuses on giving an introduction to Hopf algebra theory and a slightly unusual perspective on projective representation theory.

Chapters 3 and 4 focus on finding the equivalence classes of pointed fusion categories and the corresponding modular data. Chapter 3 introduces the category of G-graded vector spaces $\mathsf{Vec}^{\omega}G$ and explains how they are naturally pointed fusion categories. It then details the equivalence classes of these categories by proving a theorem classifying the functors between two such categories. Chapter 4 introduces the twisted quantum doubles of a finite group $D^{\omega}G$, and proves an equivalence of modular tensor categories between $\operatorname{Rep}\left(D^{\omega}G\right)$ and $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$. Using this equivalence, we give a detailed derivation for the modular data of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ and explain the link between this modular data and the modular data of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)$.

Finally, Chapters 5 and 6 focus on the computational aspects of the project. Chapter 5 describes the code written to compute the modular data of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ given G and ω. In particular it discusses the methods used to create a list of simple objects of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ and the various optimisations used to speed up the calculations of the S and T matrices from this list. Then, a database of this modular data will be assembled for all equivalence classes of pointed fusion categories with dimension at most 47. Code written by Michaël Mignard and Peter Schauenburg in their recent paper [1] is used to find a list of unitary 3-cocycles which are a representatives of a representative cohomology class of each orbit in $H^3(G, \mathbb{C})/\operatorname{Aut}(G)$.

[1] We write $2+1$ as opposed to 3 to specify that there are 2 spatial dimensions and 1 time dimension.

Then, given a group G and a unitary 3-cocycle ω my code constructs the modular data of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$. Chapter 6 will then focus on some results that can be derived from this database. In particular we confirm the result of Mignard and Schauenburg concerning the number of Morita equivalence classes of pointed fusion categories at each dimension less than 32, improve upon their lower bound when the dimension is equal to 32 and publish new lower bounds at each dimension between 33 and 47 inclusive. We note that the improvement at dimension 32 also proves that the modular data of a category is a stronger invariant than the Frobenius-Schur indicators and T-matrix.

The classification of functors $\mathsf{Vec}^{\alpha}G \to \mathsf{Vec}^{\beta}G$ in Chapter 3 is technically new but presumably obvious to experts. Cain Edie-Michell is about to publish a classification of equivalences of graded categories [4]. My result is neither a special case as it classifies more than just the equivalence classes, nor as strong as it only covers the case when the trivial graded piece is Vec.

A careful derivation of the equivalence $\text{Rep}\left(D^{\omega}G\right) \cong \mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ has not been published before. In particular, no-one seems to have noticed that with the usual conventions you are required to invert both the associator and the braiding. While the formulas presented for the S and T matrices do appear in the literature [5], the corresponding derivations of these formulas do not. These derivations are given in careful detail in Section 4.3. The database of modular data constructed as described in Chapter 5 is available for public use at `https://tqft.net/web/research/students/AngusGruen`. Such a database has never been available before. This database already been put to use by one of my fellow honours students AnRan Chen computing invariants for torus knots [6] and we are planning to use the database for numerous other computations.

Preliminaries

More in-depth discussions on the definitions in this section can be found in the textbook Tensor Categories [7].

2.1 Monoidal Category Theory

Definition 2.1. *A monoidal category* $(\mathscr{C}, \otimes, I, \alpha, \lambda, \rho)$ *consists of a category* $\mathscr{C}$ *along with a bifunctor* $\otimes : \mathscr{C} \times \mathscr{C} \to \mathscr{C}$*, an identified object* I*, and three natural isomorphisms* α*,* λ*, and* ρ*. The associator* α*, has components*

$$\alpha_{X,Y,Z} : (X \otimes Y) \otimes Z \xrightarrow{\sim} X \otimes (Y \otimes Z) \quad \forall\, X, Y, Z \in \mathscr{C}$$

and the left and right unitors λ *and* ρ*, have components*

$$\begin{aligned} \lambda_X &: I \otimes X \xrightarrow{\sim} X \\ \rho_X &: X \otimes I \xrightarrow{\sim} X \quad \forall\, X \in \mathscr{C}. \end{aligned}$$

These natural transformations satisfy the following commutative diagrams for all $W, X, Y, Z \in \mathscr{C}$*.*

$$\begin{array}{ccccc} & & (W \otimes X) \otimes (Y \otimes Z) & & \\ & \nearrow^{\alpha_{W\otimes X,Y,Z}} & & \searrow^{\alpha_{W,X,Y\otimes Z}} & \\ ((W \otimes X) \otimes Y) \otimes Z & & & & W \otimes (X \otimes (Y \otimes Z)) \\ \downarrow{\scriptstyle \alpha_{W,X,Y}\otimes 1_Z} & & & & \uparrow{\scriptstyle 1_W\otimes\alpha_{X,Y,Z}} \\ (W \otimes (X \otimes Y)) \otimes Z & & \xrightarrow{\quad\alpha_{W,X\otimes Y,Z}\quad} & & W \otimes ((X \otimes Y) \otimes Z) \end{array} \tag{2.1}$$

$$\begin{array}{ccc} (X \otimes I) \otimes Y & \xrightarrow{\quad\alpha_{X,I,Y}\quad} & X \otimes (I \otimes Y) \\ & {\scriptstyle \rho_X\otimes 1_Y}\searrow \quad \swarrow{\scriptstyle 1_X\otimes\lambda_Y} & \\ & X \otimes Y & \end{array} \tag{2.2}$$

If all three of α, λ, and ρ correspond to the identity natural transformation, then the category is called strict. A monoidal category is an example of a process known as categorification. Recall that a monoid consists of a set S, with an function $* : S \times S \to S$ satisfying $(a * b) * c = a * (b * c)$ for all $a, b, c \in S$ such that there exists a unique element $e \in S$ such that $a * e = e * a = a$ for every $a \in S$. Categorification takes a set theoretic definition and replaces sets by categories, functions by functors and equations by natural transformations satisfying certain commutative diagrams. Then a monoidal category is a categorification of a monoid. Similarly, many of the following definitions will be categorifications of more well known structures.

Theorem 2.2 (Mac Lane's Coherence Theorem)**.** *In a monoidal category, any diagram with edges consisting only of tensor products of α, λ, ρ, and identity morphisms is commutative.*

Proof. See [8]. □

Given two monoidal categories $(\mathscr{C}, \otimes, I, \alpha, \lambda, \rho)$ and $(\mathscr{D}, \cdot, J, \beta, \kappa, \mu)$, a monoidal functor between them is as follows.

Definition 2.3. *A monoidal functor (F, Φ, ϕ) consists of a functor $F : \mathscr{C} \to \mathscr{D}$ along with a natural isomorphism called the tensorator,*

$$\Phi_{X,Y} : F(X) \cdot F(Y) \xrightarrow{\sim} F(X \otimes Y) \quad \forall\, X, Y \in Ob(\mathscr{C})$$

and an isomorphism

$$\phi \in \mathscr{D}(J \xrightarrow{\sim} F(I)).$$

There are three commutative diagrams that these transformations must satisfy.

$$\begin{array}{ccc} (F(X) \cdot F(Y)) \cdot F(Z) & \xrightarrow{\beta_{F(X),F(Y),F(Z)}} & F(X) \cdot (F(Y) \cdot F(Z)) \\ \downarrow{\scriptstyle \Phi_{X,Y} \otimes 1_Z} & & \downarrow{\scriptstyle 1_X \otimes \Phi_{Y,Z}} \\ (F(X \otimes Y)) \cdot F(Z) & & F(X) \cdot (F(Y \otimes Z)) \\ \downarrow{\scriptstyle \Phi_{X \otimes Y, Z}} & & \downarrow{\scriptstyle \Phi_{X, Y \otimes Z}} \\ F((X \otimes Y) \otimes Z) & \xrightarrow{F(\alpha_{X,Y,Z})} & F(X \otimes (Y \otimes Z)) \end{array} \tag{2.3}$$

$$\begin{array}{ccc} J \cdot F(X) & \xrightarrow{\phi \otimes 1_X} & F(I) \cdot F(X) \\ \downarrow{\scriptstyle \kappa_{F(X)}} & & \downarrow{\scriptstyle \Phi_{I,X}} \\ F(X) & \xleftarrow[F(\lambda_X)]{} & F(I \otimes X) \end{array} \tag{2.4}$$

$$\begin{array}{ccc} F(X) \cdot J & \xrightarrow{1_X \otimes \phi} & F(X) \cdot F(I) \\ \downarrow{\scriptstyle \mu_{F(X)}} & & \downarrow{\scriptstyle \Phi_{X,I}} \\ F(X) & \xleftarrow[F(\rho_X)]{} & F(X \otimes I) \end{array} \tag{2.5}$$

These three commutative diagrams are each relating a structure in $\mathscr{C}$ with the corresponding structure in $\mathscr{D}$. Commutative Diagram 2.3 relates the two associators and the other two commutative diagrams relate the left unitors and right unitors respectively. This definition is a categorification of a monoid homomorphism. Unlike monoids, there is another layer that can be placed on top of monoidal functors. Given two monoidal functors (F, Φ, ϕ) and (G, Θ, θ) from $(\mathscr{C}, \otimes, I, \alpha, \lambda, \rho)$ to $(\mathscr{D}, \cdot, J, \beta, \kappa, \mu)$, there can be monoidal natural transformations between them.

Definition 2.4. *A monoidal natural transformation η, is a natural transformation from $F \to G$ which satisfies the following commutative diagrams.*

$$\begin{array}{ccc} F(X) \cdot F(Y) & \xrightarrow{\eta_X \otimes \eta_Y} & G(X) \cdot G(Y) \\ \downarrow{\scriptstyle \Phi_{X,Y}} & & \downarrow{\scriptstyle \Theta_{X,Y}} \\ F(X \otimes Y) & \xrightarrow{\eta_{X \otimes Y}} & G(X \otimes Y) \end{array} \tag{2.6}$$

$$\begin{array}{ccc} & J & \\ {}^{\phi}\swarrow & & \searrow^{\theta} \\ F(I) & \xrightarrow{\eta_I} & G(I) \end{array} \tag{2.7}$$

In general, most of the monoidal categories that we work with have additional structures or properties on top of just a monoidal structure. For this thesis we will need to define a modular tensor category. Expanding this prefix, a modular tensor category is a finitely semisimple pivotal braided linear monoidal category with a simple unit and an invertible S matrix. The next couple of subsections will be devoted to explaining each of these prefixes.

2.1.1 Pivotal and Braided Categories

Let $\mathscr{C}$ be a monoidal category.

Definition 2.5. *A right dual of an object $X \in \mathscr{C}$ is an object $X^* \in \mathscr{C}$ with an evaluation map*

$$\epsilon_X : X^* \otimes X \to I$$

and a co-evaluation map

$$\eta_X : I \to X \otimes X^*.$$

These maps must satisfy the condition that compositions[1]

$$X \xrightarrow{\eta_X \otimes 1_X} X \otimes X^* \otimes X \xrightarrow{1_X \otimes \epsilon_X} X$$
$$X^* \xrightarrow{1_{X^*} \otimes \eta_X} X^* \otimes X \otimes X^* \xrightarrow{\epsilon_X \otimes 1_{X^*}} X^*$$

are both the identity.

While right duals are not unique, they are unique up to unique isomorphism [7]. There is a related notion of left dual, ${}^*X$ which can be elegantly defined as an object in $\mathscr{C}$ which has X as a right dual.

Definition 2.6. *The category $\mathscr{C}$ is a rigid monoidal category if every object has both a left and a right dual.*

In a rigid category, for any object X it should be clear that ${}^*X^* \cong X$.

It is also possible to take the dual of a morphism. Given a morphism $f \in \mathscr{C}(X \to Y)$, f^* is a morphism in $\mathscr{C}(Y^* \to X^*)$ defined by the composition

$$Y^* \xrightarrow{1_{Y^*} \otimes \eta_X} Y^* \otimes X \otimes X^* \xrightarrow{1_{Y^*} \otimes f \otimes 1_{X^*}} Y^* \otimes Y \otimes X^* \xrightarrow{\epsilon_Y \otimes 1_{X^*}} X^*.$$

The morphism ${}^*f$ is similarly defined as

$${}^*Y \xrightarrow{\eta_{*X} \otimes 1_{*Y} \otimes} {}^*X \otimes X \otimes {}^*Y \xrightarrow{1_{*X} \otimes f \otimes 1_{*Y}} {}^*X \otimes Y \otimes {}^*Y \xrightarrow{1_{*X} \otimes \epsilon_{*Y}} {}^*X.$$

Observe that the two conditions on compositions given in the definition of * are exactly stating that $(1_X)^* = 1_{X^*}$ and ${}^*(1_X) = 1_{*X}$. There are two features of these definition that I will state without proof here. These proofs are a little verbose currently but will become almost trivial when using string diagrams which will be introduced in the next section.

[1] We implicitly assume that the monoidal category $\mathscr{C}$ is strict for this definition. There is a similar but more complicated condition when $\mathscr{C}$ is not strict.

Lemma 2.7. *The dual operator on morphisms satisfies* ${}^*f^* = f$ *and* $(f \circ g)^* = g^* \circ f^*$.

Another property of the dual operator is that it flips monoidal products. If X and Y are objects with right duals X^* and Y^* respectively then $(X \otimes Y)^* \cong Y^* \otimes X^*$. The evaluation and co-evaluation maps are given by $\epsilon_{X \otimes Y} = \epsilon_Y \circ (1 \otimes \epsilon_X \otimes 1)$ and $\eta_{X \otimes Y} = (1 \otimes \eta_X \otimes 1) \circ \eta_Y$ respectively.

This leads to the following important lemma.

Lemma 2.8. *The dual operator is an op-monoidal contravariant functor.*

Briefly, a contravariant functor $F : \mathscr{C} \to \mathscr{D}$ is a mapping between two categories which flips morphisms. This means that for each $X \in \mathscr{C}$, $F(X)$ is an object in $\mathscr{D}$ but, unlike a regular functor[2], if $f \in \mathscr{C}(X \to Y)$ then $F(f) \in \mathscr{C}(F(Y) \to F(X))$. This mapping is required to satisfy the pair of equations $F(1_X) = 1_{F(X)}$ and $F(f \circ g) = F(g) \circ F(f)$. Then, an op-monoidal functor is almost a monoidal functor except the op-tensorator is a natural transformation from $F(Y) \cdot F(X) \xrightarrow{\sim} F(X \otimes Y)$.

The proof of Lemma 2.8 is immediate from Lemma 2.7 and the discussions preceding and ensuing it.[3]

A slightly stronger notion than duals is that of invertible objects.

Definition 2.9. *An object X is said to be invertible if $X^* \otimes X \cong X \otimes X^* \cong I$.*

Clearly if this is the case, X^* is also invertible and $X \cong X^{**}$. More generally, it is often useful to require an isomorphism between X and its double dual. This requires extra structure on the category called a pivotal isomorphism.

Definition 2.10. *A pivotal monoidal category is a pair $(\mathscr{C}, \phi)$ with $\mathscr{C}$ a rigid monoidal category and ϕ a monoidal natural isomorphism from the identity functor to the double dual functor, [9].*

Observe that the composition of two op-monoidal contravariant functors is a monoidal functor. Hence the double dual functor is indeed a monoidal functor from $\mathscr{C}$ to itself and so ϕ is well defined. The pivotal isomorphism also gives an isomorphism between X^* and ${}^*X$. Therefore, in a pivotal category X^* can be unambiguously referred to as the dual of X.

Definition 2.11. *A braided monoidal category $(\mathscr{C}, \sigma)$ consists of a monoidal category $\mathscr{C}$ with a natural isomorphism*

$$\sigma_{X,Y} : X \otimes Y \to Y \otimes X$$

such that the following diagrams commute.

$$\begin{array}{ccccc} (X \otimes Y) \otimes Z & \xrightarrow{\alpha_{X,Y,Z}} & X \otimes (Y \otimes Z) & \xrightarrow{\sigma_{X,Y \otimes Z}} & (Y \otimes Z) \otimes X \\ \downarrow{\scriptstyle \sigma_{X,Y} \otimes 1_Z} & & & & \downarrow{\scriptstyle \alpha_{Y,Z,X}} \\ (Y \otimes X) \otimes Z & \xrightarrow{\alpha_{Y,X,Z}} & Y \otimes (X \otimes Z) & \xrightarrow{1_Y \otimes \sigma_{X,Z}} & Y \otimes (Z \otimes X) \end{array} \tag{2.8}$$

$$\begin{array}{ccccc} X \otimes (Y \otimes Z) & \xrightarrow{\alpha^{-1}_{X,Y,Z}} & (X \otimes Y) \otimes Z & \xrightarrow{\sigma_{X \otimes Y,Z}} & Z \otimes (X \otimes Y) \\ \downarrow{\scriptstyle 1_X \otimes \sigma_{Y,Z}} & & & & \downarrow{\scriptstyle \alpha^{-1}_{Z,X,Y}} \\ X \otimes (Z \otimes Y) & \xrightarrow{\alpha^{-1}_{X,Z,Y}} & (X \otimes Z) \otimes Y & \xrightarrow{\sigma_{X,Z} \otimes 1_Y} & (Z \otimes X) \otimes Y \end{array} \tag{2.9}$$

A braiding on a monoidal category is a categorification of commutativity.

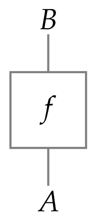
[2] These are commonly called covariant functors when in situations where contravariant functors also appear.

[3] Once again we have implicitly assumed that $\mathscr{C}$ is strict. Everything remains true if $\mathscr{C}$ is not a strict monoidal category but all the proofs become more complicated.

2.1.2 String Diagrams

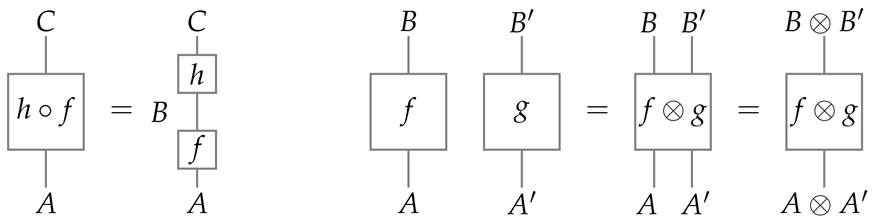
String diagrams are a method of representing information in a category in terms of a graphical calculus of planar diagrams. By convention, strings are read going up the page.

Definition 2.12. *Given a morphism $f : A \to B$ its string diagram is*



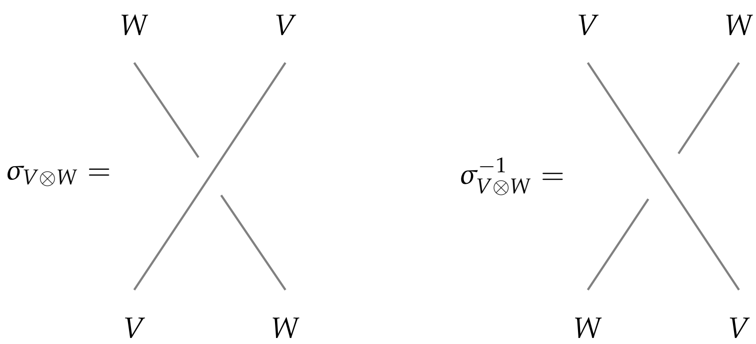
points correspond to objects and morphisms correspond to boxes on lines connecting two points.

Composition of morphisms f, h is vertical stacking and the tensor product is horizontal juxtaposition.



There is a slight abuse of notation here. For a category with no extra structure, a string diagram is really referring to the equivalence class of string diagrams up to a certain planar isotopies. The allowed isotopies change as more structures and properties are added to the categories. Currently, we are allowed to slide the morphism boxes around as if they were beads on the string and move the strings so long as the end points are fixed, no part of a string is ever horizontal and strings do not cross each other.

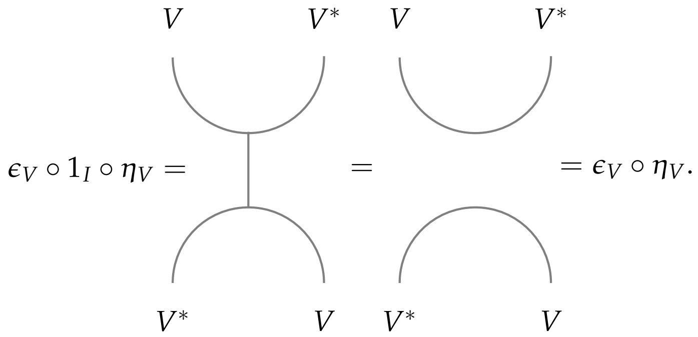
Various special morphisms are assigned particular diagrams. A simple example is that an empty string, with no boxes, crossings, or horizontal tangents corresponds to the identity morphism. Often an identity morphism is unambiguously referred to as the object itself. More interesting examples occur when the category has additional structures. For example, if the category is braided then the braiding and inverse braiding are represented by over and under crossings diagrams.



In a braided category, the equivalence class of string diagrams is now up to planar isotopy that includes moving part of a string over/under another string to create or remove over/under crossings. We are not however allowed to pass strings through each other.

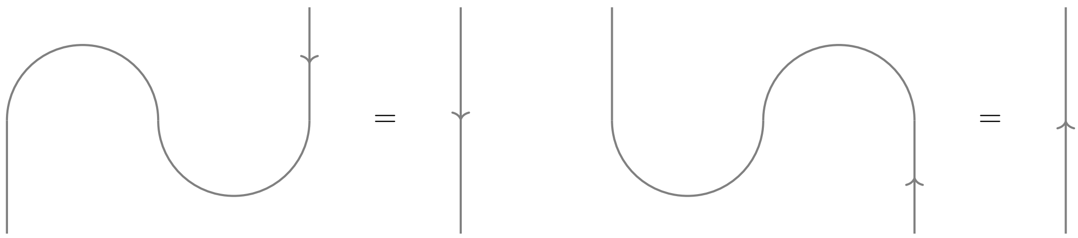
If the category is rigid then the evaluation and co-evaluation morphisms are represented by the following strings.

Note that in this picture, the line corresponding to 1_I as well as the point I have been replaced by empty space. When the unitors in the category are trivial,[4] this can be done with no loss of information. This serves to simplify diagrams, the morphism $\epsilon_V \circ \eta_V$ is



Another problem is what to do with associators. When the category is strict, they can be safely ignored, however this will not be possible for this thesis. On the other hand including them will massively clutter the string diagrams. Therefore, they will be omitted from string diagrams throughout and this will mean that some care will need to be taken reading morphisms directly off string diagrams. Note that by Theorem 2.2, any two morphisms read off the same string diagram are equal.[5]

When the category is rigid, there is also a convenient way to represent dual objects. This is by assigning directions to the lines with the convention that a line travelling up the page corresponds to the object and one travelling down the page corresponds to its right dual. Then, the the conditions on the ϵ and η correspond to the following isotopies.

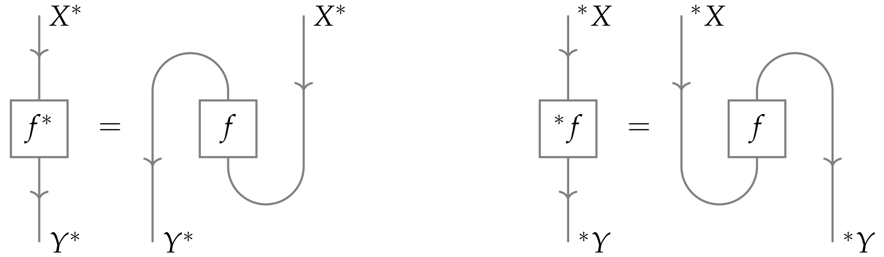


As a brief justification for the use of string diagrams let us go back and prove Lemma 2.7.

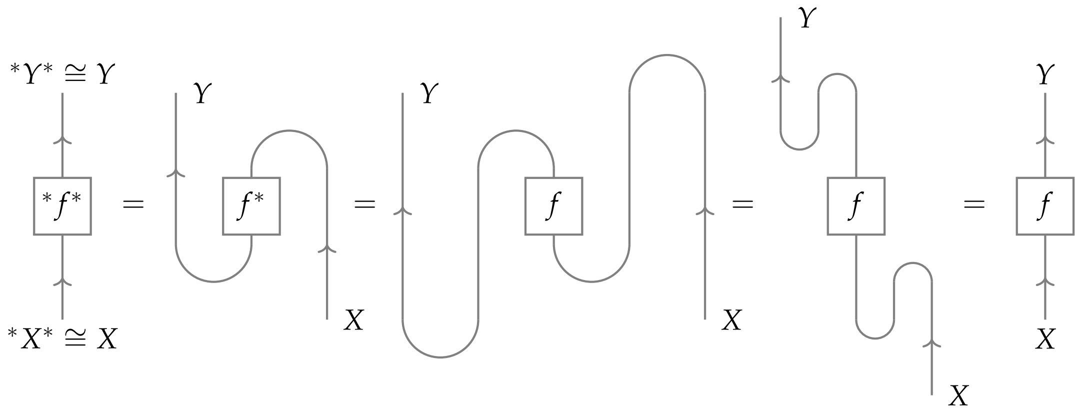
[4]Which they will be for this thesis.

[5]We need to be a little careful here. This holds provided the initial and final bracketing of the objects is unchanged for both morphisms

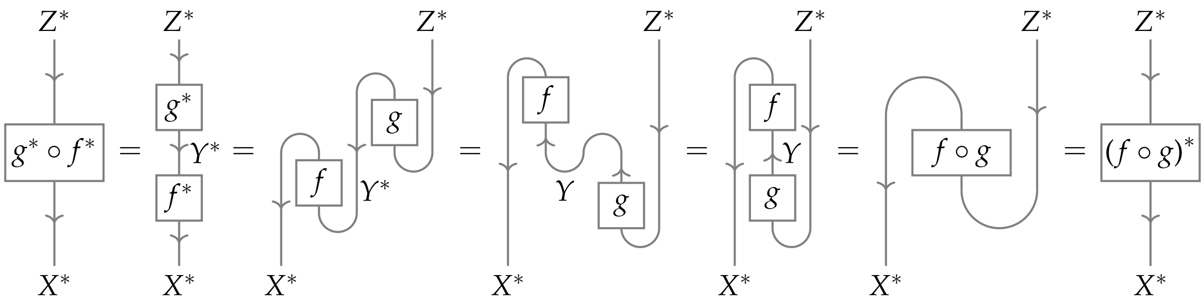
Proof. Observe that the two morphisms f^* and ${}^*f$ are defined by the following strings



Then, the following isotopies of string diagrams shows that ${}^*f^* = f$.

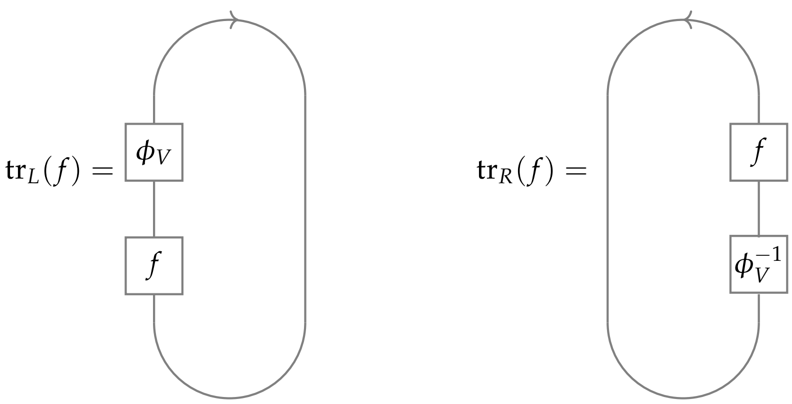


Similarly, different isotopies prove that $(f \circ g)^* = g^* \circ f^*$



∎

If the category is pivotal, the pivotal isomorphism allows for the creation of closed loops. This gives rise to the definition of a trace. Given a morphism $f \in \mathscr{C}(V \to V)$ define the left and right traces of f to be



These are morphisms in $\mathscr{C}(I \to I)$. Commonly, $\mathscr{C}(I \to I)$ will be isomorphic to a field $\mathbb{K}$ and when this is the case, a trace corresponds to an evaluation map from $\mathscr{C}(V \to V) \to \mathbb{K}$. In Vec, the category of finite dimensional vector spaces with morphisms linear maps, the trace of a morphism is exactly the trace of the matrix representing the morphism after a basis is chosen.

Definition 2.13. *A pivotal monoidal category is spherical if all string diagrams behave as if they where on the surface of a sphere, [9]. Equivalently,* $\mathrm{tr}_L(f) = \mathrm{tr}_R(f)$ *for all* f*.*

In particular in a spherical category, strings can be pulled around the back of the sphere which identifies the two possible traces.[6] In a spherical category, the dimension of an object is defined to be the trace of the identity.

$$\dim(V) = \mathrm{tr}(1_V)$$

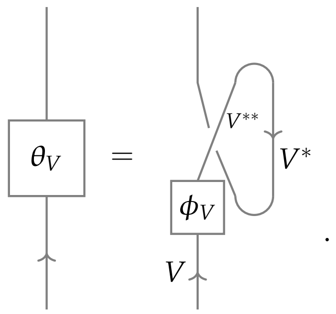
Combining all the structure and properties that have been defined so far brings us to ribbon categories.

Definition 2.14. *A ribbon category is a braided spherical monoidal category.*

These categories possess a twist map θ_V for each object satisfying

$$\theta_{V\otimes W} = \sigma_{V,W}\sigma_{W,V}(\theta_V \otimes \theta_W).$$

This map is defined by the string diagram



The proof that this morphism satisfies the alleged condition is easily shown using string manipulations and so will be omitted. The reason that these are called ribbon categories is because, if one-dimensional strings are replaced by ribbons embedded in 3-space, θ can be represented by a full twist in the ribbon.

2.1.3 Fusion and Modular Categories

Definition 2.15. *Given a field* $\mathbb{K}$*, a* $\mathbb{K}$*-linear category is a category* $\mathscr{C}$ *whose morphism spaces* $\mathscr{C}(X \to Y)$ *are* $\mathbb{K}$*-vector spaces with the composition map being bilinear.*

Any $\mathbb{K}$-algebra is naturally a $\mathbb{K}$-linear category with a single object. More generally, a $\mathbb{K}$-linear category is equivalent to a $\mathbb{K}$-algebroid. A linear functor is a functor whose action on morphism spaces corresponds to a linear map. All functors whose source and target are linear categories will be assumed to be linear functors. If an endomorphism space $\mathrm{End}(X) = \mathscr{C}(X \to X)$ is one dimensional, it must be generated by the identity morphism 1_X and so, as $F(1_X) = 1_{F(X)}$, the action of a linear functor F on $\mathrm{End}(X)$ is determined by its action on X.

[6] Pulling a string around the back of a sphere will possibly invert ϕ or have other small effects like sending η_V to η_{V^*}, these minor changes can be read directly off the string diagram.

Definition 2.16. *An initial object is an object $X \in \mathscr{C}$ such that for all $Y \in \mathscr{C}$, $\mathscr{C}(X \to Y)$ contains only one element. Similarly, X is a terminal object if for all $Y \in \mathscr{C}$, $\mathscr{C}(Y \to X)$ contains only one element.*

Lemma 2.17. *Initial and terminal objects are unique up to isomorphism.*

Proof. This is a standard result in category theory, a proof can be found in [8] for the interested reader. □

An object that is both initial and terminal is said to be a zero object and by Lemma 2.17 this is also unique up to isomorphism. In a $\mathbb{K}$-linear category, these conditions exactly require that $\mathscr{C}(X \to Y)$ is the 0 dimensional vector space, $\{0\}$.

Definition 2.18. *A $\mathbb{K}$-linear category with a 0 object is semisimple if there exists a collection of objects $\{X_i\}_{i \in I}$ called the simple objects with the following set of properties:*

1. *For all i and j in I,*

$$\mathscr{C}(X_i \to X_j) = \begin{cases} \mathbb{K} & \text{if } i = j \\ \{0\} & \text{otherwise.} \end{cases}$$

2. *All finite coproducts exists.*

3. *Any object $X \in \mathscr{C}$ is isomorphic to a (possibly infinite or null) direct sum of simple objects.*

An object which is isomorphic to a finite direct sum of simple objects is called finitely semisimple. A semisimple category is said to be finitely semisimple if every object is finitely semisimple and there are only finitely many isomorphism classes of simple objects.

When a category is semisimple this often allows for analysis of the category to be restricted to merely analysing the simple objects. This comes from the fact that many of the structures that can be placed on a category respect these decompositions. For example, if a semisimple category has a monoidal structure, then the associator will satisfy

$$\alpha_{W \oplus X,Y,Z} = \alpha_{W,Y,Z} \oplus \alpha_{X,Y,Z}.$$

This feature of semisimple categories will be implicitly used throughout this thesis as it will massively simplify proofs. Additionally, functors whose sources and targets are semisimple categories will also respect such decompositions

$$F(X \oplus Y) = F(X) \oplus F(Y).$$

Definition 2.19. *A fusion category is a finitely semisimple rigid linear monoidal category such that the unit object I is simple.*

This thesis will focus on a particular subclass of fusion categories called pointed fusion categories.

Definition 2.20. *A pointed fusion category is a fusion category such that every simple object is invertible.*

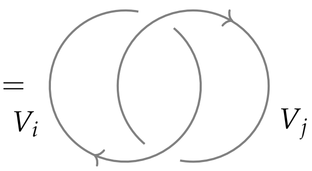
The rank of a fusion category rank($\mathscr{C}$) is the number of isomorphism classes of simple objects. If $\mathscr{C}$ is a pivotal fusion category then the left and right traces are maps from $\mathscr{C}(V \to V)$ to $\mathbb{K}$ as $\mathscr{C}(I \to I)$ is identified with $\mathbb{K}$ via the identification $1_I \mapsto 1$. If $\mathscr{C}$ is spherical then we can define a dimension of the category. To find $\dim(\mathscr{C})$, choose a representative V_i from each isomorphism class of simple objects and then[7]

$$\dim(\mathscr{C}) = \sum_i \dim(V_i)^2.$$

If $\mathscr{C}$ is a ribbon category then there is a matrix we can construct by taking traces of the double braiding applied to pairs of simple objects.

Definition 2.21. *Let $\mathscr{C}$ be a ribbon fusion category with simple objects $\{V_i\}$ for $i \in \{1, \cdots, n\}$. Define the $\tilde{S}$ matrix by the string diagram*

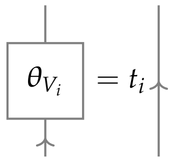
$\tilde{S}_{i,j} =$



Then $\mathscr{C}$ is a modular tensor category if $\tilde{S}$ is invertible.

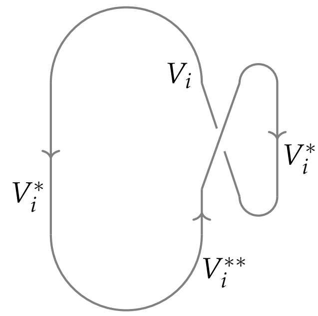
The matrix $\tilde{S}$ is also known as the unnormalised S-matrix. The normalised S-matrix, S, is equal to $\frac{1}{\sqrt{\dim(\mathscr{C})}}\tilde{S}$.

There is a second matrix that can be created from a modular tensor category. By definition, for every simple object V_i, $\mathscr{C}(V_i \to V_i) = \mathbb{K}$. Therefore, the twist θ_V is some multiple of the identity id_{V_i} defined as t_i.



Then, the T matrix is defined to be the diagonal matrix with i'th diagonal element equal to t_i. Taking a trace, T can be calculated by evaluating the string diagram

$T_{i,j} = \frac{\delta_{i,j}}{\dim(V_i)}$



In general, the matrix produced by this would also need to be normalised however, in this thesis we only deal with categories where the normalisation factor is 1. For a modular tensor category $\mathscr{C}$, the pair of matrices S, T is known as the modular data.

The reason that it is called modular data is because it gives a representation of the modular group $\mathrm{SL}(2, \mathbb{Z})$ by way of the map

$$\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \mapsto S \quad \text{and} \quad \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \mapsto T.$$

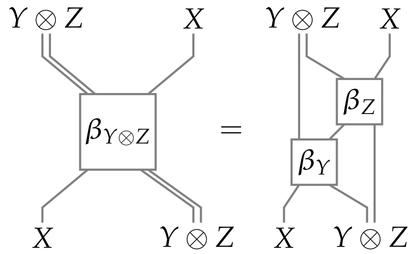
[7]It is an easy proof that this definition is independent of the chosen representatives; isomorphic objects have equal dimensions.

This means that the modular data must satisfy the relations $S^2 = (ST)^3$ and $S^4 = I$ and in practise it also satisfies more complex relations [10] which I will discuss in more detail in Chapter 5.

2.1.4 The Drinfeld centre

The Drinfeld centre [11] is an operation that takes a monoidal category $\mathscr{C}$ and produces a braided monoidal category denoted $\mathcal{Z}(\mathscr{C})$. Often $\mathcal{Z}(\mathscr{C})$ is referred to as the centre of $\mathscr{C}$. The construction of the Drinfeld centre is as follows.

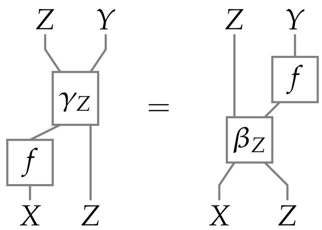
Let $(\mathscr{C}, \otimes, I, \alpha, \lambda, \rho)$ be a monoidal category. Fixing any object $X \in \mathscr{C}$, a half braiding for X is a natural isomorphism $\beta : X \otimes \_ \xrightarrow{\sim} \_ \otimes X$ such that



Writing this out as a condition on morphisms, the condition is

$$\alpha^{-1}_{Y,Z,X} \circ (1 \otimes \beta_Z) \circ \alpha_{Y,X,Z} \circ (\beta_Y \otimes 1) \circ \alpha^{-1}_{X,Y,Z} = \beta_{Y\otimes Z}. \tag{2.10}$$

Then, objects of $\mathcal{Z}(\mathscr{C})$ are pairs (X, β) of X an object in $\mathscr{C}$ and β a half braiding for X. Morphisms in $\mathcal{Z}(\mathscr{C})$ are morphisms in $\mathscr{C}$ that commute with the half braidings. Explicitly, $f \in \mathcal{Z}(\mathscr{C})\big((X,\beta) \to (Y,\gamma)$ if and only if $f \in \mathscr{C}(X \to Y)$ and



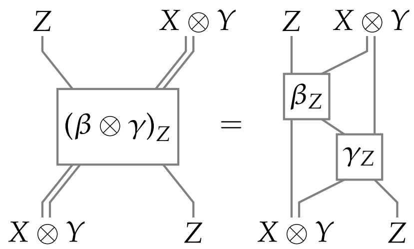
for all $Z \in \mathscr{C}$. The monoidal structure on $\mathcal{Z}(\mathscr{C})$ is[8]

$$(X,\beta) \otimes (Y,\gamma) = (X \otimes Y, \alpha_{\_,X,Y} \circ (\beta \otimes 1) \circ \alpha^{-1}_{X,\_,Y} \circ (1 \otimes \gamma) \circ \alpha_{X,Y,\_}). \tag{2.11}$$

The identity object is simply $(I, \lambda \circ \rho^{-1})$ and the left and right unitors and associator come directly from the category $\mathscr{C}$. Observe that the action of the tensor product on half braidings is

[8] The blank sections are where the object to be twisted should appear. E.g. if given a $Z \in \mathscr{C}$ the natural transformation is $\alpha_{Z,X,Y} \circ (\beta_Z \otimes 1) \circ \alpha^{-1}_{X,Z,Y} \circ (1 \otimes \gamma_Z) \circ \alpha_{X,Y,Z}$

exactly given by the string diagram



Theorem 2.22. *The Drinfeld centre of a category can be made into a braided monoidal category.*

Proof. The easiest way to prove this theorem is to produce a braiding. Hence define the braiding σ by

$$\sigma_{(X,\beta),(Y,\gamma)} = \beta_Y.$$

Note that

$$\sigma'_{(X,\beta),(Y,\gamma)} = \gamma_X^{-1} = \sigma^{-1}_{(Y,\gamma),(X,\beta)}$$

is another possible braiding. *A priori* neither of these braidings is the correct one to ascribe and, as we will observe later, in our particular case we will use the inverse braiding. Note that $\mathcal{Z}(\mathscr{C})$ will refer to the regular braiding and $\mathcal{Z}(\mathscr{C})^{bop}$ will refer to the inverse braiding.[9]

It is of course required to show that these braidings both satisfy the relevant commutative diagrams, namely Diagrams 2.8 and 2.9. But, these follow directly from the fact that half braidings satisfy Equation 2.10 and by the definition of how the tensor product acts on half braidings given in Equation 2.11. □

An important aspect of the Drinfeld centre is that it preserves both the monoidal properties of $\mathscr{C}$ as well as any added monoidal structures on $\mathscr{C}$.

Theorem 2.23. *If $\mathscr{C}$ is rigid, pivotal, spherical or fusion then so is $\mathcal{Z}(\mathscr{C})$. In particular if $\mathscr{C}$ is a spherical fusion category then $\mathcal{Z}(\mathscr{C})$ is a modular tensor category.*

The proof of this theorem is outside the scope of this thesis but can be found in [12] for an interested reader. In particular it is important to note that both rigidity and pivotality can be directly induced by the corresponding properties and structures on $\mathcal{Z}(\mathscr{C})$. Interestingly, while it is certainly true that equivalent categories have equivalent centres, the converse is not true.

Definition 2.24. *Two monoidal categories $\mathscr{C}$ and $\mathscr{D}$ are said to be Morita equivalent if $\mathcal{Z}(\mathscr{C}) \cong \mathcal{Z}(\mathscr{D})$.*

Note that Morita equivalence is a weaker condition than equivalence of categories this is shown explicitly by Figure 6.2. It should also be noted that there is a relationship between the dimension of the category and the dimension of its centre. Indeed we have the simple formula

$$\dim(\mathcal{Z}(\mathscr{C})) = \dim(\mathscr{C})^2.$$

There is not however a well understood relationship between the rank of a category and the rank of its centre. This will be explored briefly in Chapter 6.

[9]bop stands for opposite braiding and $\mathscr{C}^{bop}$ is a well defined category whenever $\mathscr{C}$ is braided. We can similarly define $\mathscr{C}^{op}$ and $\mathscr{C}^{mop}$ which correspond to reversing all morphisms and the tensor product respectively. These adjectives can be stacked leading to mouthfuls like $\mathscr{C}^{op,mop,bop}$.

2.2 Backgroup Algebra

2.2.1 Group Cohomology

There is a small amount of group cohomology that will be useful later and so is given here.

Let G be a finite group with identity e and let $\mathbb{K}$ be an algebraically closed field. Define $C^n(G,\mathbb{K})$ to be the group under pointwise multiplication of all functions from $G^n \to \mathbb{K}^\times$. Note that these functions are not *a priori* required to interact with G's group structure. Define the maps $\partial^{n+1} : C^n(G,\mathbb{K}) \to C^{n+1}(G,\mathbb{K})$ by

$$\begin{aligned}\partial^{n+1}(\phi)(g_1,\cdots,g_{n+1}) = \phi(g_2,\cdots,g_{n+1})& \\ \times \prod_{i=1}^{n} \phi(g_1,\cdots,g_{i-1},g_i g_{i+1},g_{i+2},\cdots,g_{n+1})^{(-1)^i}& \\ \times\, \phi(g_1,\cdots,g_n)^{(-1)^{n+1}}.&\end{aligned} \tag{2.12}$$

Theorem 2.25. *The sequence*

$$\cdots \longleftarrow C^{n+1} \xleftarrow{d^{n+1}} C^n \xleftarrow{d^n} C^{n-1} \longleftarrow \cdots \tag{2.13}$$

is a cochain complex.

This is a standard result in group cohomology, see [13].

Define $Z^n(G,\mathbb{K}) = \ker(d^{n+1})$, $B^n(G,\mathbb{K}) = \operatorname{Im}(d^n)$ and $H^n(G,\mathbb{K}) = Z^n(G,\mathbb{K})/B^n(G,\mathbb{K})$. Elements of $C^n(G,\mathbb{K})$ are known as n-cochains, elements of $Z^n(G,\mathbb{K})$ as n-cocycles and elements of $B^n(G,\mathbb{K})$ as n-coboundaries.

Definition 2.26. *Two cocycles, α, β, are cohomologous if $\alpha\beta^{-1}$ is a coboundary. A cocycle is cohomologically trivial if it is cohomologous to the* 1 *cocycle or equivalently if it is a coboundary.*

Definition 2.27. *A normalized cochain is a cochain which outputs* 1 *whenever any of its inputs are e.*

Definition 2.28. *A normalized cochain α is unitary if $\alpha^{|G|} = 1$.*

This leads up to the theorem that will be needed later on.

Theorem 2.29. *Every cocycle is cohomologous to a unitary cocycle.*

Proof. See Appendix A. □

2.2.2 Hopf Algebras

For any finite group G its representations form a category known as $\operatorname{Rep}(G)$. The objects of this category are representations of G and the morphisms are intertwining maps. Moreover, there is a natural equivalence between $\operatorname{Rep}(G)$ and $\operatorname{Rep}(\mathbb{K}[G])$, the category of representations of the group algebra $\mathbb{K}[G]$. It is well known that $\operatorname{Rep}(G)$ carries much more structure than merely a category. Indeed, $\operatorname{Rep}(G)$ is a ribbon fusion category whose simple objects correspond to the irreducible representations of G. A natural question then is if this extra structure on $\operatorname{Rep}(G) \cong \operatorname{Rep}(\mathbb{K}[G])$ can be visualised down in the algebra $\mathbb{K}[G]$. Indeed it can be and corresponds to added structures on $\mathbb{K}[G]$ such as co-multiplication, and an antipode. The general term for algebras with these two extra structures is a Hopf algebra and representations of Hopf Algebras naturally form rigid monoidal categories.

If a category can be shown to be equivalent to a representation category of a Hopf algebra then often this vastly simplifies the study of that category. In particular, categorical calculations can be replaced by algebraic ones which allow for a wealth more methods to simplify and solve them. The following sections will give an introduction to Hopf algebras and their representation categories. For more discussion on the structures presented here, see [14].

Let $\mathbb{K}$ be an algebraically closed field and assume for the rest of the section that all maps are $\mathbb{K}$ linear and all tensor products are over $\mathbb{K}$. Initially let H be a $\mathbb{K}$ vector space.

Definition 2.30. *An algebra is a triple (H, η, ∇) where*

$$\eta : \mathbb{K} \to H$$
$$\nabla : H \otimes H \to H$$

satisfy the commutative diagrams.

$$\begin{array}{ccc} H \otimes H \otimes H & \xrightarrow{\nabla \otimes 1} & H \otimes H \\ \downarrow{\scriptstyle 1 \otimes \nabla} & & \downarrow{\scriptstyle \nabla} \\ H \otimes H & \xrightarrow{\nabla} & H \end{array} \qquad \begin{array}{ccc} \mathbb{K} \otimes H \cong H \cong H \otimes \mathbb{K} & \xrightarrow{1 \otimes \eta} & H \otimes H \\ \downarrow{\scriptstyle \eta \otimes 1} & \searrow^{1} & \downarrow{\scriptstyle \nabla} \\ H \otimes H & \xrightarrow{\nabla} & H \end{array}$$

The first commutative diagram is precisely forcing the multiplication map to be associative and the second commutative diagram is forcing a multiplicative identity, $\eta(1)$.

Define $F : H_1 \otimes H_2 \to H_2 \otimes H_1$ to be the flip map $F(x \otimes y) = y \otimes x$. An algebra is commutative if $\nabla = \nabla \circ F$. Given two algebras, $(H_1, \eta_1, \nabla_1), (H_2, \eta_2, \nabla_2)$, there is a natural algebra structure on $H_1 \otimes H_2$ given by $\big(H_1 \otimes H_2, \eta_1 \otimes \eta_2, (\nabla_1 \otimes \nabla_2) \circ (1 \otimes F \otimes 1)\big)$.

Moving on, a coalgebra is defined by reversing every arrow in the definition of an algebra.

Definition 2.31. *A coalgebra is a triple (H, ϵ, Δ) where*

$$\epsilon : H \to \mathbb{K}$$
$$\Delta : H \to H \otimes H$$

satisfy the commutative diagrams

$$\begin{array}{ccc} H & \xrightarrow{\Delta} & H \otimes H \\ \downarrow{\scriptstyle \Delta} & & \downarrow{\scriptstyle 1 \otimes \Delta} \\ H \otimes H & \xrightarrow{\Delta \otimes 1} & H \otimes H \otimes H \end{array} \qquad \begin{array}{ccc} H & \xrightarrow{\Delta} & H \otimes H \\ \downarrow{\scriptstyle \Delta} & \searrow^{1} & \downarrow{\scriptstyle 1 \otimes \epsilon} \\ H \otimes H & \xrightarrow{\epsilon \otimes 1} & \mathbb{K} \otimes H \cong H \cong H \otimes \mathbb{K} \end{array}$$

Similarly to the algebra case, the first commutative diagram represents coassociativity and, cocommutativity occurs if $\Delta = F \circ \Delta$. Additionally, if $(H_1, \epsilon_1, \Delta_1)$ and $(H_2, \epsilon_2, \Delta_2)$ are coalgebras then the tensor product coalgebra will be $\big(H_1 \otimes H_2, \epsilon_1 \otimes \epsilon_2, (1 \otimes F \otimes 1) \circ (\Delta_1 \otimes \Delta_2)\big)$.

A bialgebra is a vector space with algebra and a coalgebra structures (η, ∇) and (ϵ, Δ) such that both η and ∇ are coalgebra homomorphisms and both ϵ and Δ are algebra homomorphisms.

Definition 2.32. *A bialgebra is a tuple $(H, \eta, \nabla, \epsilon, \Delta)$ such that (H, η, ∇) is an algebra, (H, ϵ, Δ) is a coalgebra and the following four diagrams commute.*

$$\begin{array}{ccccc} H \otimes H & \xrightarrow{\nabla} & H & \xrightarrow{\Delta} & H \otimes H \\ \downarrow{\scriptstyle \Delta \otimes \Delta} & & & & \uparrow{\scriptstyle \nabla \otimes \nabla} \\ H \otimes H \otimes H \otimes H & & \xrightarrow{1 \otimes F \otimes 1} & & H \otimes H \otimes H \otimes H \end{array} \tag{2.14}$$

$$\begin{array}{ccc} \mathbb{K} & \xrightarrow{\eta} & H \\ & \searrow^{1} & \downarrow \epsilon \\ & & \mathbb{K} \end{array} \tag{2.15}$$

$$\begin{array}{ccc} H \otimes H & \xrightarrow{\qquad \nabla \qquad} & H \\ & \searrow^{\epsilon \otimes \epsilon} \quad \swarrow_{\epsilon} & \\ & \mathbb{K} \otimes \mathbb{K} \cong \mathbb{K} & \end{array} \tag{2.16}$$

$$\begin{array}{ccc} & \mathbb{K} \otimes \mathbb{K} \cong \mathbb{K} & \\ & \swarrow_{\eta \otimes \eta} \quad \searrow^{\eta} & \\ H \otimes H & \xleftarrow[\Delta]{\qquad\qquad} & H \end{array} \tag{2.17}$$

The tensor product structure on algebras and coalgebras defined earlier extends to bialgebras in the obvious way.

A Hopf algebra is a bialgebra with an extra map called an antipode.

Definition 2.33. *A Hopf algebra is a pair (H, S) where H is a bialgebra and S an automorphism of H such that the following diagram commutes*

$$\begin{array}{ccccc} H \otimes H & & \xrightarrow{\quad S \otimes 1 \quad} & & H \otimes H \\ \Delta \uparrow & & & & \downarrow \nabla \\ H & \xrightarrow{\epsilon} & \mathbb{K} & \xrightarrow{\eta} & H \\ \downarrow \Delta & & & & \nabla \uparrow \\ H \otimes H & & \xrightarrow{\quad 1 \otimes S \quad} & & H \otimes H \end{array} \tag{2.18}$$

The tensor product of two Hopf algebras (H_1, S_1) and (H_2, S_2) is the Hopf algebra $(H_1 \otimes H_2, S_1 \otimes S_2)$. In general it is too much to ask for Hopf algebras to be cocommutative but there is a weaker notion known as almost cocommutative that uses the algebra structure on $H \otimes H$.

Definition 2.34. *A Hopf algebra H, is almost cocommutative if there exists an invertible element $R \in H \otimes H$ such that*

$$\Delta^{op}(h) = F \circ \Delta(h) = R\Delta(h)R^{-1} \tag{2.19}$$

for all $h \in H$.

Note that ∇ has been omitted in the above equations for clarity and for the same reason will be omitted from here on out.

Define the maps $\phi_{ij} : H \otimes H \to H \otimes H \otimes H$ by

$$\phi_{12}(a \otimes b) = a \otimes b \otimes 1$$
$$\phi_{13}(a \otimes b) = a \otimes 1 \otimes b$$
$$\phi_{23}(a \otimes b) = 1 \otimes a \otimes b.$$

Additionally, define $R_{ij} = \phi_{ij}(R)$.

Definition 2.35. *A quasitriangular Hopf algebra is an almost cocommutative Hopf algebra such that*

$$(\Delta \otimes 1)(R) = R_{13}R_{23} \tag{2.20}$$
$$(1 \otimes \Delta)(R) = R_{13}R_{12}. \tag{2.21}$$

It can also occur that the coproduct is not coassociative. Again if there is an algebra structure it is possible to weaken coassociativity to allow for it to fail up to conjugation by an invertible element.

Definition 2.36. *A quasi bialgebra is a tuple $(H, \epsilon, \Delta, \Phi, l, r)$ with H an algebra and Φ is an invertible element of $H \otimes H \otimes H$ satisfying the following equations for every $h \in H$.*

$$(\Delta \otimes 1) \circ \Delta(h) = \Phi\big((1 \otimes \Delta) \circ \Delta(h)\big)\Phi^{-1} \tag{2.22}$$

$$\Big[(1 \otimes 1 \otimes \Delta)(\Phi)\Big]\Big[(\Delta \otimes 1 \otimes 1)(\Phi)\Big] = (1 \otimes \Phi)\Big[(1 \otimes \Delta \otimes 1)(\Phi)\Big](\Phi \otimes 1) \tag{2.23}$$

$$(\epsilon \otimes 1)(\Delta(h)) = l^{-1}hl \tag{2.24}$$

$$(1 \otimes \epsilon)(\Delta(h)) = r^{-1}hr \tag{2.25}$$

$$\Big(1 \otimes \epsilon \otimes 1\Big)(\Phi) = 1 \otimes 1 \tag{2.26}$$

For the purposes of this thesis assume that $l = r = 1$. This will simplify some proofs later on. In particular it makes the unitors in the representation category of a quasi bialgebra trivial. Also note that Φ^{-1} will also satisfy Equations 2.23 and 2.26.

Definition 2.37. *A quasi Hopf algebra is a tuple (H, S, α, β) with H a quasi bialgebra, $\alpha, \beta \in H$ and $S : H \to H$ satisfying the following conditions. Let $\Delta(h) = \sum_i h_{i_1} \otimes h_{i_2}$, $\Phi = \sum_i X_{i_1} \otimes X_{i_2} \otimes X_{i_3}$ and $\Phi^{-1} = \sum_i Y_{i_1} \otimes Y_{i_2} \otimes Y_{i_3}$, then we require that*

$$\sum_i X_{i_1}\beta S(X_{i_2})\alpha X_{i_3} = 1$$

$$\sum_i S(Y_{i_1})\alpha Y_{i_2}\beta S(Y_{i_3}) = 1$$

and for all $h \in H$,

$$\sum_i S(h_{i_1})\alpha h_{i_2} = \epsilon(h)\alpha$$

$$\sum_i h_{i_1}\beta S(h_{i_2}) = \epsilon(h)\beta.$$

The quasitriangular structure can also be extended to quasi Hopf algebras.

Definition 2.38. *A quasitriangular quasi Hopf algebra is a tuple (H, R) with H a quasi Hopf algebra and R and invertible element in $H \otimes H$ satisfying*

$$\Delta^{op}(h) = F \circ \Delta(h) = R\Delta(h)R^{-1}$$

for all $h \in H$ and

$$(\Delta \otimes 1)R = \Phi_{321}R_{13}\Phi^{-1}_{132}R_{23}\Phi_{123} \tag{2.27}$$

$$(1 \otimes \Delta)R = \Phi^{-1}_{231}R_{13}\Phi_{213}R_{12}\Phi^{-1}_{123} \tag{2.28}$$

where $\Phi_{abc} = \sum_i X_{i_a} \otimes X_{i_b} \otimes X_{i_c}$ and $\Phi_{123} = \Phi = \sum_i X_{i_1} \otimes X_{i_2} \otimes X_{i_3}$

2.2.3 Representation Theory of Hopf Algebras

Definition 2.39. *A representation of an algebra H is a vector space V with an algebra homomorphism $\rho : H \to \mathrm{End}_{\mathbb{K}}(V)$.*

This ρ can either be thought of as a map $H \to (V \to V)$ or a map $H \times V \to V$. With the second interpretation, a representation is exactly the same as a left module. Note that, ρ being an algebra homomorphism, means that $\rho(h_1) \circ \rho(h_2) = \rho\big(\nabla(h_1, h_2)\big)$ and for all $k \in \mathbb{K}$, and $v \in V$, $\rho(\eta(k), v) = kv$.

Let $\mathrm{Rep}(H)$ be the category whose objects are representations of H and morphisms are intertwining maps. Equivalently, from the above interpretation, $\mathrm{Rep}(H)$ can be thought of as the category whose objects are left modules with morphisms module homomorphisms. These two points of view will be used interchangeably.

Theorem 2.40. *If H is a quasitriangular quasi Hopf algebra then* $\mathrm{Rep}(H)$ *is a rigid braided monoidal category.*

While ideally this thesis would prove the above theorem, in practise the proof would be far too lengthy. Instead we will prove the following three lemmas which combine to a slightly weaker theorem. It should however be clear in the proofs of these lemma's how they could be extended to stronger statements so as to prove the above theorem.

Lemma 2.41. *If H is a quasi bialgebra then* $\mathrm{Rep}(H)$ *is a monoidal category.*

Lemma 2.42. *If H is a Hopf algebra then the antipode S gives* $\mathrm{Rep}(H)$ *the property of a rigid monoidal category.*

Lemma 2.43. *If H is a quasitriangular Hopf algebra then the quasitriangular element induces a braiding on* $\mathrm{Rep}(H)$.

Note that Lemmas 2.42 and 2.43 assume that the underlying coalgebra is coassociative.

Proof of Lemma 2.41. Let (U, ρ), (V, σ) and (W, τ) be representations of H. Initially we must construct the tensor product of representations. Clearly $U \otimes V$ is a representation of $H \otimes H$ via the map $\rho \otimes \sigma$. In order to make this a representation of H, recall that the coproduct Δ is an algebra homomorphism. Hence define $(U, \rho) \otimes (V, \sigma) = \big(U \otimes V, (\rho \otimes \sigma) \circ \Delta\big)$. An easy calculation checks that if $f : U \to W$ and $g : V \to X$ are intertwining maps, then $f \otimes g : U \otimes V \to W \otimes X$ is also an intertwining map.

The identity object is exactly $(\mathbb{K}, \epsilon)$ and the isomorphisms $(\mathbb{K}, \epsilon) \otimes (V, \sigma) \xrightarrow{\sim} (V, \sigma)$ and $(V, \sigma) \otimes (\mathbb{K}, \epsilon) \xrightarrow{\sim} (V, \sigma)$ follow from Equations 2.24 and 2.25 after the simplification that $l = r = 1$.

This leaves us to define the associator. The two possible tensor products of the three representations are[10]

$$\big((U, \rho) \otimes (V, \sigma)\big) \otimes (W, \tau) = \big(U \otimes V \otimes W, (\rho \otimes \sigma \otimes \tau) \circ (\Delta \otimes 1) \circ \Delta\big)$$

and

$$(U, \rho) \otimes \big((V, \sigma) \otimes (W, \tau)\big) = \big(U \otimes V \otimes W, (\rho \otimes \sigma \otimes \tau) \circ (1 \otimes \Delta) \circ \Delta\big)$$

Recall that there is an identified element $\Phi \in H \otimes H \otimes H$ satisfying

$$(\Delta \otimes 1) \circ \Delta(h) = \Phi\big((1 \otimes \Delta) \circ \Delta(h)\big)\Phi^{-1}.$$

Therefore,

$$(\rho \otimes \sigma \otimes \tau)(\Phi^{-1}) \circ (\rho \otimes \sigma \otimes \tau)\big((\Delta \otimes 1) \circ \Delta(h)\big) = (\rho \otimes \sigma \otimes \tau)\Big(\Phi^{-1}\big((\Delta \otimes 1) \circ \Delta(h)\big)\Big)$$

[10] Note that we are working here with a model of Vec where the monoidal structure is strict. If our model of Vec had a non strict monoidal structure this theorem would still certainly be true, all the formulas would simply become slightly more complicated.

$$= (\rho\otimes\sigma\otimes\tau)\Big(\big((1\otimes\Delta)\circ\Delta(h)\big)\Phi^{-1}\Big)$$
$$= (\rho\otimes\sigma\otimes\tau)\big((1\otimes\Delta)\circ\Delta(h)\big)\circ(\rho\otimes\sigma\otimes\tau)(\Phi^{-1}).$$

This means that $\alpha_{\rho,\sigma,\tau} = (\rho\otimes\sigma\otimes\tau)(\Phi^{-1})$ is an intertwining map

$$\alpha_{\rho,\sigma,\tau} : \Big((U,\rho)\otimes(V,\psi)\Big)\otimes(W,\phi)\to(U,\rho)\otimes\Big((V,\psi)\otimes(W,\phi)\Big).$$

Moreover, Equation 2.23 directly forces α to satisfy Commutative Diagram 2.1 and, as the unitors are trivial Equation 2.26 shows that α will satisfy Commutative Diagram 2.2. Therefore α is an and so with these structures, $(\mathrm{Rep}(H),\otimes,(\mathbb{K},\epsilon),\alpha,1,1)$ is a monoidal category. □

It is unfortunate that in the definition of α, Φ^{-1} is used instead of Φ. This is the commonly used notation in the field but will cause some trouble later on. Moving on, we next wish to prove that for a Hopf algebra H, $\mathrm{Rep}(H)$ is rigid.

Proof of Lemma 2.42. Note that a bialgebra is equivalent to a quasi bialgebra with $\Phi = 1\otimes1\otimes1$ and $l = r = 1$. Thus, the previous proof gives us a strict monoidal structure on $\mathrm{Rep}(H)$.

Let (V,ρ) be an object in $\mathrm{Rep}(H)$. Define a new object (V^*,ρ^*) where V^* is the vector space dual of V and, for all $h\in H$, $f\in V^*$ and $m\in V$,

$$\rho^*(h,f)(m) = f\big(\rho(S(h),m)\big).$$

Claim: This constructed object (V^*,ρ^*) is the dual object of (V,ρ).

Let γ_V be the usual linear map $V^*\otimes V\to\mathbb{K}$ and recall that the action of H on $V^*\otimes V$ and $\mathbb{K}$ is given by $(\rho^*\otimes\rho)\circ\Delta$ and ϵ respectively. Then the following computation shows that γ_V commutes with the action of H.

$$\begin{aligned}
\gamma_V\circ\big((\rho^*\otimes\rho)\circ\Delta(h)\big)(f\otimes m) &= \sum_i \gamma_V\circ\big((\rho^*\otimes\rho)(h_{i_1}\otimes h_{i_2})(f\otimes m)\\
&= \sum_i \gamma_V\Big(\rho^*(h_{i_1},f)\otimes\rho(h_{i_2},m)\Big)\\
&= \sum_i \rho^*(h_{i_1},f)\Big(\rho(h_{i_2},m)\Big)\\
&= \sum_i f\Big(\rho\big(S(h_{i_1}),\rho(h_{i_2},m)\big)\Big)\\
&= f\circ\rho\circ(S\otimes\rho)\circ(\Delta\otimes1)(h,m)\\
&= f\circ\rho\circ(1\otimes\rho)\circ(S\otimes1\otimes1)\circ(\Delta\otimes1)(h,m)\\
&= f\circ\rho\circ(\nabla\otimes1)\circ(S\otimes1\otimes1)\circ(\Delta\otimes1)(h,m) && *\\
&= f\circ\rho\circ\big((\eta\circ\epsilon)\otimes1\big)(h,m) && **\\
&= \epsilon(h)f(m) && ***\\
&= \epsilon(h)\circ\gamma_V(f\otimes m)
\end{aligned}$$

Note that the steps $*$ and $***$ are simple using the definition of ρ as an algebra homomorphism and in particular in $***$ observing that $\epsilon(h)\in\mathbb{K}$. Then, step $**$ is a direct consequence of Commutative Diagram 2.18. In order for (V^*,ρ^*) to be the dual, there also needs to be a map from $(\mathbb{K},\epsilon)\to(V,\rho)\otimes(V^*,\rho^*)$. Again, we can use the regular vector space map ζ_V, which is, for a basis $\{v_i\}$ of V with corresponding basis $\{v_i^*\}$ of V^*

$$\zeta_V(1) = 1\sum_i v_i\otimes v_i^*.$$

A calculation no worse than the above will show that ζ_V commutes with the action of H. This shows that every element has a right dual. An identical argument will also show that every element must have a left dual and so $\mathrm{Rep}(H)$ is a rigid monoidal category. □

In the case where H is only a quasi Hopf algebra, the α and β in Definition 2.37 increase the complexity of the evaluation and co-evaluation maps. For the right dual of a left module (V,ρ), the evaluation map γ_V is replaced by $\gamma_V \circ (1\otimes\rho(\alpha))$ and the co-evaluation map ζ_V will be replaced by $(1\otimes\rho^*(\beta))\circ\zeta_V$. Similar modifications will be made to the regular vector space structures for left duals. Next, consider the case when the the Hopf algebra is quasitriangular. This will allow us to assign a braiding to $\mathrm{Rep}(H)$.

Proof of Lemma 2.43. Let H be a quasitriangular Hopf algebra. This means that there is an invertible $R\in H\otimes H$ such that $F\circ\Delta(h) = R\Delta(h)R^{-1}$ and R satisfies Equations 2.20 and 2.21.

Let (V,ρ) and (W,τ) be elements of $\mathrm{Rep}(H)$ with the strict monoidal structure defined earlier. Then, a braiding will be an isomorphism between $\big(V\otimes W, (\rho\otimes\tau)\circ\Delta\big)$ and $\big(W\otimes V, (\tau\otimes\rho)\circ\Delta\big)$. Using the flip map F, we have $\forall h\in H,\ w\in W,\ v\in V$ the equality

$$\begin{aligned}\big((\tau\otimes\rho)\circ\Delta(h)\big)(w,v) &= F\circ\Big((\rho\otimes\tau)\circ F\circ\Delta(h)\Big)\circ F(w,v)\\ &= F\circ(\rho\otimes\tau)(R\Delta(h)R^{-1})\circ(w,v)\\ &= F\circ(\rho\otimes\tau)(R)\circ(\rho\otimes\tau)(\Delta(h))\circ\big(F\circ(\rho\otimes\tau)(R)\big)^{-1}(w,v).\end{aligned}$$

Rewriting this gives

$$F\circ(\rho\otimes\tau)(R)\circ(\rho\otimes\tau)(\Delta(h))\circ(v,w) = \big((\tau\otimes\rho)\circ\Delta(h)\big)\circ F\circ(\rho\otimes\tau)(R)(v,w). \tag{2.29}$$

This shows that $\sigma_{\rho,\tau} = F\circ(\rho\otimes\tau)(R)$ is an isomorphism

$$\sigma_{\rho,\tau} : \big(V\otimes W, (\rho\otimes\tau)\circ\Delta\big) \xrightarrow{\sim} \big(W\otimes V, (\tau\otimes\rho)\circ\Delta\big).$$

In order for σ to be a braiding it needs to satisfy Commutative Diagrams 2.8 and 2.9 (These give conditions on how the braiding needs to interact with the tensor product and associators.).

Let (U,ψ), (V,ρ) and (W,τ) be three objects in $\mathrm{Rep}(H)$. Then, as the associators are trivial, Commutative Diagram 2.8 requires the equality

$$(1\otimes\sigma_{\psi,\tau})\circ(\sigma_{\psi,\rho}\otimes 1) = \sigma_{\psi,(\rho\otimes\tau)\circ\Delta}$$

Equation 2.21 states that

$$(1\otimes\Delta)R = R_{13}R_{12} = (F\otimes 1)\Big(1\otimes R\Big)\times(R\otimes 1).$$

Therefore,

$$\begin{aligned}\sigma_{\psi,(\rho\otimes\tau)\circ\Delta} &= F\circ\Big(\psi\otimes\big((\rho\otimes\tau)\circ\Delta\big)\Big)(R)\\ &= (1\otimes F)\circ(F\otimes 1)\circ\Big(\psi\otimes\rho\otimes\tau)(1\otimes\Delta)\Big)(R)\\ &= (1\otimes F)\circ(F\otimes 1)\circ(\psi\otimes\rho\otimes\tau)(R_{13}R_{12})\\ &= (1\otimes F)\circ(F\otimes 1)\circ(\psi\otimes\rho\otimes\tau)(R_{13})\circ(\psi\otimes\rho\otimes\tau)(R_{12})\\ &= (1\otimes F)\circ(F\otimes 1)\circ\big((\psi\otimes\rho\otimes\tau)\circ(F\otimes 1)(1\otimes R)\big)\circ(\psi\otimes\rho\otimes\tau)(R\otimes 1)\\ &= (1\otimes F)\circ(\rho\otimes\psi\otimes\tau)(1\otimes R)\circ(F\otimes 1)\circ(\psi\otimes\rho\otimes\tau)(R\otimes 1)\end{aligned}$$

$$= \Big(1 \otimes \big(F \circ (\psi \otimes \tau)(R)\big)\Big) \circ (F \otimes 1) \circ \Big(\big(F \circ (\psi \otimes \rho)(R)\big) \otimes 1\Big)$$
$$= (1 \otimes \sigma_{\psi,\tau}) \circ (\sigma_{\psi,\rho} \otimes 1).$$

The key step in this proof is the observation that

$$(F \otimes 1) \circ \big((\psi \otimes \rho \otimes \tau) \circ (F \otimes 1)(1 \otimes R)\big) = (\rho \otimes \psi \otimes \tau)(1 \otimes R) \circ (F \otimes 1).$$

The idea is essentially that $(F \otimes 1)$ can be pulled through the action $\big((\psi \otimes \rho \otimes \tau) \circ (F \otimes 1)\big)(1 \otimes R)\big)$ and act upon each component permuting $(\psi \otimes \rho \otimes \tau) \mapsto (\rho \otimes \psi \otimes \tau)$ and $(F \otimes 1)(1 \otimes R) \mapsto (F \otimes 1) \circ (F \otimes 1)(1 \otimes R) = (1 \otimes R)$.

Therefore as required, σ satisfies Diagram 2.8. A very similar calculation using Equation 2.20 will show that σ also satisfies Diagram 2.9. For brevity this calculation will be skipped.

Therefore σ is indeed a valid braiding and so $(\text{Rep}(H), \sigma)$ is a braided tensor category. □

In the more general case when H is a quasitriangular quasi Hopf algebra, Equations 2.20 and 2.21 will be replaced by Equations 2.27 and 2.28. Looking at these new equations, it should be clear that they are exactly taking into account the non trivial cocycle. Hence, in the general case the braiding will still be given by σ as defined above.

2.2.4 Projective Representation Theory

This section is based on the paper "A Character Theory for Projective Representations of Finite Groups" by Chuangxun Cheng [15].

Let G be a finite group, let V be a complex vector space and let β be a unitary 2-cocycle. Recall $\text{GL}(V)$ and $\text{PGL}(V)$; the first is the ring $\text{End}(V)$ and the second is $\text{GL}(V)/\mathbb{C}$. Where $\mathbb{C}$ is naturally identified with the subring of $\text{End}(V)$ of multiples of the identity map.

Definition 2.44. *A β-projective representation is a pair (V, ρ) with V a vector space and ρ a function $\rho : G \to \text{GL}(V)$ satisfying*

$$\rho(g)\rho(h) = \beta(g, h)\rho(gh)$$

As β is a 2-cocycle ρ is associative but ρ is clearly not a group homomorphism when $\beta \neq 1$. This is a slightly unusual definition of a projective representation. In general, a projective representation is defined as a group homomorphism $G \to \text{PGL}(V)$. For our purposes this second definition is unsatisfactory because we care about the value of the 2-cocycle β. When mapped through to $\text{PGL}(V)$, constants disappear and as such there would be no difference between different values of β.

As an example of the difference between these perspectives, consider the case when $G = \mathbb{Z}/2\mathbb{Z} = \{0, 1\}$. If we view projective representations as a map to $\text{PGL}(V)$, the only one dimensional projective representation is the trivial one. However, there are non trivial one dimensional β-projective representations. For example when β is given by $\beta(1, 1) = -1$ and all other pairs map to 1, two projective representations are given by $1 \mapsto i$ and $1 \mapsto -i$.

Much of classical representation theory carries across to projective representations. This includes the following definitions and results, stated here without proof. Proofs of these theorems can be found in [15].

Definition 2.45. *Two projective representations (V, ρ), (W, ψ) are linearly equivalent if there exists an isomorphism $f : V \to W$ such that $f \circ \rho(g) = \psi(g) \circ f$ for all $g \in G$.*

There is another coarser notion of equivalence called projective equivalence. It is essentially the same definition but the equality occurs in $\text{PGL}(W)$ instead of $\text{GL}(W)$. The two

β-projective representations of $\mathbb{Z}/2\mathbb{Z}$ given a moment ago are examples of representations that are projectively but not linearly equivalent.

Definition 2.46. *Given two β-projective representations (V, ρ), (W, ψ), we can form a new β-projective representation $(V \oplus W, \rho \oplus \psi)$ called the direct sum.*

Theorem 2.47. *Let (V, ρ) be a β-projective representation and $W \subset V$ a G invariant subspace. Then (V, ρ) can be decomposed into the direct sum of two β-projective representations, $(W, \rho|_W)$ and $(W', \rho|_{W'})$ where $W \oplus W' = V$ and $\rho|_W$ represents the action of ρ on V restricted to W.*

Definition 2.48. *A projective representation (V, ρ) is said to be irreducible if the only G invariant subspaces are V and $\{0\}$.*

Theorem 2.49. *All projective representations can be decomposed into a direct sum of irreducible representations. This decomposition is unique up to reordering and linear equivalences.*

Definition 2.50. *Given a projective representation, the character χ_ρ of the representation is defined to be*

$$\chi_\rho(g) = \operatorname{tr}\left(\rho(g)\right).$$

Theorem 2.51. *A β-projective representation is entirely determined by its character.*

Note that unlike classical representation theory, $\chi_\rho(g)$ need not be constant across conjugacy classes. Indeed a simple calculation shows that for a β-projective representation

$$\begin{aligned}
\chi_\rho(hgh^{-1}) &= \operatorname{tr}\left(\rho(hgh^{-1})\right) \\
&= \frac{1}{\beta(h,g)\beta(hg,h^{-1})} \operatorname{tr}\left(\rho(h)\rho(g)\rho(h^{-1})\right) \\
&= \frac{1}{\beta(h,g)\beta(hg,h^{-1})} \operatorname{tr}\left(\rho(g)\rho(h^{-1})\rho(h)\right) \\
&= \frac{\beta(h^{-1},h)}{\beta(h,g)\beta(hg,h^{-1})} \chi_\rho(g).
\end{aligned}$$

Another calculation that will be useful later is the relationship between $\chi_\rho(g)$ and $\chi_\rho(g^{-1})$. In order to prove this we will first prove that the eigenvalues of $\chi_\rho(g)$ must be roots of unity. Let $n < \infty$ be the order of g. Then

$$I = \rho(1) = \rho(g^n) = \beta(g, g^{n-1})^{-1} \cdots \beta(g,g)^{-1}\rho(g)^n.$$

As β is unitary, $\beta^{|G|} = 1$ and so

$$I = I^{|G|} = \rho(g)^{n|G|}.$$

Therefore the eigenvalues of $\rho(g)$ must be roots of unity which means that the eigenvalues of $\rho(g)^{-1}$ are the complex conjugates of the eigenvalues of $\rho(g)$. Thus as $\rho(g)\rho(g^{-1}) = \alpha(g, g^{-1})I$ we find that

$$\chi_\rho(g^{-1}) = \operatorname{tr}\left(\rho(g^{-1})\right) = \operatorname{tr}(\alpha(g,g^{-1})\rho(g)^{-1}) = \alpha(g,g^{-1}) \operatorname{tr}(\rho(g)^{-1}) = \alpha(g,g^{-1})\chi_\rho^*(g^{-1}), \tag{2.30}$$

where χ^* denotes the complex conjugate of χ.

Usefully, it turns out that it is possible to visualise β-projective representations as true linear representations of a larger group. The following theorem will be essential later on when we try and compute the β-projective character table for a given group and 2-cocycle.

Theorem 2.52. *Let G be a group and β a unitary 2-cocycle. Then there exists a group G_β called the group extension of G and an injective map f from β-projective representations of G to linear representations of G_β.*

Proof. First to construct G_β. As β is unitary it takes values in the group of $|G|$'th roots of unity in $\mathbb{C}$. This group is naturally identified with $A = C_{|G|} = \langle x | x^{|G|} \rangle$ by the isomorphism $e^{\frac{2\pi i}{|G|}} \mapsto x$. Let $\tilde{\beta}$ be image of β under the isomorphism into A. As a set, $G_\beta = G \times A$ and its group structure comes from a twisted multiplication map

$$(g, x^n) \times (h, x^m) = (gh, \tilde{\beta}(g, h)x^{n+m}).$$

Associativity of the multiplication follows from the associativity of the multiplication in G and A as well as the 2-cocycle condition which β satisfies. As β is normalized, there is an identity element $(e, 1)$, and inverses are given by $(g, x^m)^{-1} = (g^{-1}, \left(\beta(g, g^{-1})\right)^{-1} x^{-m}$. This proves that G_β is indeed a group.

Note that A is isomorphic to the normal subgroup of G_β generated by (e, x) and that G is not a subgroup of G_β but is a quotient group given by $G = G_\beta / A$.

Next let (V, ρ) be some β-projective representation of G. Define $f(\rho) : G_\beta \to GL(V)$ by

$$f(\rho)\big(g, x^m\big) = e^{\frac{2m\pi i}{|G|}} \rho(g).$$

The following calculation shows that $(V, f(\rho))$ is indeed a linear representation of G_β.

$$\begin{aligned} f(\rho)\big(g, x^m\big) f(\rho)\big(h, x^n\big) &= e^{\frac{2(m+n)\pi i}{|G|}} \rho(g)\rho(h) \\ &= \beta(g, h) e^{\frac{2(m+n)\pi i}{|G|}} \rho(gh) \\ &= f(\rho)\big(gh, \tilde{\beta}(g, h)x^{n+m}\big) \\ &= f(\rho)\big((g, m) \times (h, m)\big). \end{aligned}$$

The injectivity of f is immediate from noting that $f(\rho)\big(g, 1\big) = \rho(g)$ and this completes the proof of Theorem 2.52. □

An important corollary of this theorem is as follows.

Corollary 2.53. *The β-projective representation (V, ρ) will be irreducible if and only if $(V, f(\rho))$ is an irreducible linear representation.*

Proof. Let W be a G_β invariant subspace of V. Then as $f(\rho)\big(g, 1\big) = \rho(g)$, W must also be a G invariant subspace. Conversely, assume that W is not a G_β invariant subspace. Then there exists some element $w \in W$ and $(g, x^m) \in G_\beta$ such that $f(\rho)\big(g, x^m\big)\big)(w) \notin W$. Therefore $\rho(g)(w) = e^{\frac{-2m\pi}{|G|}} f(\rho)\big(g, x^m\big)(w) \notin W$ and so W is not a G invariant subspace. Therefore G invariant subspaces of V correspond to G_β invariant subspace of V. □

Importantly for us, when looking at β-irreducible representations this map f has a well behaved inverse on $\mathrm{Im}(f)$. It is also easy to check if an irreducible representation of G_β is in $\mathrm{Im}(f)$.

Let (V, ρ) be an irreducible linear representation of G_β. As $A \cong \langle (e, x) \rangle$ is in the center of G_β and $x^{|A|} = 1$, Schur's Lemma gives us that $\rho(e, x) = e^{\frac{2k\pi i}{|A|}} I_n$ for some integer k. Then define $f'(\rho)(g) = \rho(g, 1)$. Note that $f'(f(\rho')) = \rho'$ and so

$$f'(\rho)(g) f'(\rho)(h) = \rho(g, 1)\rho(h, 1) = \rho(gh, \tilde{\beta}(g, h)) = \beta(g, h)^k \rho(gh, 1) = \beta(g, h)^k f'(\rho(gh)).$$

Clearly, f' will be a β-projective representation if and only if $k = 1$ and, when restricted to such representations $f(f'(\rho)) = \rho$. Additionally, there turns out to be an easy check to see if $k = 1$ using characters, as this is equivalent to asking if $\frac{\chi(e,x)}{\chi(e,1)} = e^{\frac{2\pi i}{|A|}}$.

This gives us a method of producing the β-character table of G only using linear representations. First we construct G_β and find its character table. Then we create a subtable of all characters that satisfy $\frac{\chi(1,x)}{\chi(e,1)} = e^{\frac{2\pi i}{|A|}}$. Restricting this subtable to G, Corollary 2.53 shows that this is exactly the β-character table of G. This exact method has been encoded into GAP and the code will be displayed in chapter 5.

One final observation is that, as with linear representations, β-projective representations can also be viewed as left modules of an algebra. When $\beta = 1$, it is well known that representations of G are equivalent to left modules of $\mathbb{K}[G]$. Recall that elements of $\mathbb{K}[G]$ are sums $\sum_{g\in G} a_g g$ with each a_g some constant in $\mathbb{K}$ and multiplication is defined by

$$\Big(\sum_{g\in G} a_g g\Big)\Big(\sum_{h\in G} b_h h\Big) = \sum_{k\in G}\Big(\sum_{gh=k} a_g b_h\Big)k.$$

This is exactly the linearised version of multiplication in G. More generally, β-projective representations can be viewed as left modules of the twisted group algebra $\mathbb{K}_\beta[G]$ whose underlying vector space is identical but multiplication is twisted by β. That is to say $g \times h = \beta(g,h)gh$ or for a more general element

$$\Big(\sum_{g\in G} a_g g\Big)\Big(\sum_{h\in G} b_h h\Big) = \sum_{k\in G}\Big(\sum_{gh=k} \beta(g,h) a_g b_h\Big)k.$$

The proof of the equivalence between β-projective representations of G and left modules of $\mathbb{K}_\beta[G]$ is immediate from the definitions of β-projective representations and left modules.

Chapter 3

Classifying equivalence classes of Pointed Fusion Categories

There exists the following well known theorem about pointed fusion categories.

Theorem 3.1. *All pointed fusion categories are equivalent to* $\mathsf{Vec}^{\alpha} G$ *for some group* G *and* 3*-cocycle,* α*.*

Proof. See [16]. □

This allows us to restrict the problem of studying the equivalence classes of pointed fusion categories to these concrete examples. This chapter will explain what these categories $\mathsf{Vec}^{\alpha} G$ are and prove several theorems that will allow further restrictions on the equivalence classes of these categories.

3.1 Twisted G-graded Vector Spaces

Fix a field $\mathbb{K}$, which we suppress for the rest of the chapter and let G be a finite group. A G-graded vector space V is a vector space which can be decomposed into a direct sum of vector spaces each sitting over a different element of G

$$V = \bigoplus_{g \in G} V_g.$$

Given two G-graded vector spaces V and W, a G-graded homomorphism $\psi : V \to W$ is a linear map that respects the G grading. This means that for all $g \in G$, the restriction of ψ to V_g is a linear map from $\psi_g : V_g \to W_g$.

Definition 3.2. *The category* $\mathsf{Vec}\ G$ *is a skeletal category whose objects are* G*-graded vector spaces and morphisms are* G*-graded homomorphisms.*

Lemma 3.3. *The category* $\mathsf{Vec}\ G$ *is linear and semisimple.*

Proof. Clearly $\mathsf{Vec}\ G$ is linear as there is a natural vector space structure on G-graded homomorphisms. Explicitly, for all $r, s \in \mathbb{K}$ and $\rho, \psi \in \mathsf{Vec}\ G(V \to W)$, $(r\rho + s\psi)$ is defined by

$$(r\rho + s\psi)(v) = r\rho(v) + s\psi(v).$$

Bilinearity of the composition map follows from the linearity of morphisms. To prove semisimplicity define the objects δ_g for each $g \in G$ to be the one dimensional $G-$graded vector spaces

$$(\delta_g)_h = \begin{cases} \mathbb{K} & \text{if } h = g \\ \{0\} & \text{if } h \neq g. \end{cases}$$

Observe that if $g \neq h$ then as $(\delta_g)_h = (\delta_h)_g = \{0\}$, the only element of Vec $G(\delta_g \to \delta_h)$ is 0. On the other hand, as $(\delta_g)_g = \mathbb{K}$, maps from $\delta_g \to \delta_g$ are identified by their action on $(\delta_g)_g = \mathbb{K}$. Hence $\text{End}(\delta_g) \cong \mathbb{K}$ with the identification $1_{\delta_g} \mapsto 1$. Then the composition map in $\text{End}(\delta_g)$ corresponds to multiplication in $\mathbb{K}$. Additionally, it should be clear that every G-graded vector spaces V, is isomorphic to a direct sum of these objects as

$$V \cong \bigoplus_{g \in G} \bigoplus_{i=1}^{|V_g|} \delta_g.$$

Therefore Vec G is a linear and semisimple category. □

Consider the following tensor product on Vec G. Letting $V, W \in$ Vec G, define $V \otimes W$ to be the G-graded vector space with g'th component

$$(V \otimes W)_g = \bigoplus_{hk=g} V_h \otimes W_k.$$

This tensor product is defined similarly on morphisms. In order to create a monoidal category, this proposed tensor product must admit an associator and there must exist a unit object and unital morphisms. The unit object is clearly δ_e but there is more choice in the rest of the structures. Let α be a natural isomorphism

$$\alpha_{X,Y,Z} : (X \otimes Y) \otimes Z \xrightarrow{\sim} X \otimes (Y \otimes Z).$$

As G is a group, its multiplication is associative and so

$$(\delta_g \otimes \delta_h) \otimes \delta_k = \delta_{(gh)k} = \delta_{g(hk)} = \delta_g \otimes (\delta_h \otimes \delta_k).$$

As $\text{End}(\delta_{ghk}) \cong \mathbb{K}$, for any tuple $(g, h, k) \in G \times G \times G$, $\alpha_{g,h,k}$ is[1] some scalar in $\mathbb{K}$. Hence, when restricted to only simple objects, α is exactly a map from $G \times G \times G \to \mathbb{K}^\times$. Then, setting $W = \delta_g, X = \delta_h, Y = \delta_k$ and $Z = \delta_l$, Commutative Diagram 2.1 (The commutative diagram equating the two possible maps from $(((W \otimes X) \otimes Y) \otimes Z \to W \otimes (X \otimes (Y \otimes Z)))$ is an equality in $\text{End}(\delta_{ghkl})$ which corresponds to the equality in $\mathbb{K}$

$$\alpha_{h,k,l}\alpha_{g,hk,l}\alpha_{g,h,k} = \alpha_{gh,k,l}\alpha_{g,h,kl}.$$

As α is uniquely defined by its action on the simple objects, the above equation implies that there is a bijection between possible associators and 3-cocycles in $Z^3(G, \mathbb{K})$. We can classify the possible unitors using a similar method. Setting $X = \delta_g$ and $Y = \delta_h$, Commutative Diagram 2.2 (The commutative diagram equating the two possible maps from $(X \otimes I) \otimes Y \to X \otimes Y$) corresponds to the equality

$$\rho_g = \alpha_{g,e,h}\lambda_h.$$

Fix $\lambda_e = k \in \mathbb{K}^\times$ and observe that the 3-cocycle condition on the tuple (e, e, e, e) forces that $\alpha_{e,e,e} = 1$. Then, as the above equation is satisfied for all g, h, this means that $\rho_e = \lambda_e = k$ and more generally all of the λ's and ρ's on simple objects can be written in terms of α and k

$$\rho_g = k\alpha_{g,e,e} \tag{3.1}$$

[1]For simplicity, whenever we subscript α or more generally any natural transformation by group elements, this refers to the corresponding simple object. e.g. $\alpha_{h,k,l} = \alpha_{\delta_h,\delta_k,\delta_l}$.

$$\lambda_h = \frac{k}{\alpha_{e,e,h}}. \tag{3.2}$$

While all possible choices of k produce different monoidal categories it can be easily proven that these categories are all monoidally equivalent. For α a 3-cocycle and $k \in \mathbb{K}^\times$, let $\mathsf{Vec}_k^\alpha G$ denote the monoidal category of G graded vector spaces whose associator on the simple objects corresponds to multiplication by α and unitors on the simple objects corresponds to multiplication by ρ and λ as defined in Equations 3.1 and 3.2. Additionally, define $\mathsf{Vec}^\alpha G = \mathsf{Vec}_1^\alpha G$. Note that there is a mild abuse of notation here in that α will be simultaneously referred to both as a 3-cocycle and as the associator.

Lemma 3.4. *The categories* $\mathsf{Vec}_k^\alpha G$ *and* $\mathsf{Vec}^\alpha G$ *are equivalent monoidal categories.*

Proof. Consider the monoidal functor (F, Φ, ϕ) from $\mathsf{Vec}^\alpha G$ to $\mathsf{Vec}_k^\alpha G$ with F the identity functor, Φ the identity natural transformation and ϕ multiplication by k in $\mathbb{K} \cong \text{End}(I)$. It is a simple check to confirm that these definitions satisfy Commutative Diagrams 2.3, 2.4, and 2.5. Therefore this defines a monoidal functor which is clearly invertible and so the two categories $\mathsf{Vec}_k^\alpha G$ and $\mathsf{Vec}^\alpha G$ are equivalent monoidal categories. □

The following theorem will be stated without proof; the proof is provided later on in the chapter as a more complete understanding of the monoidal equivalence classes of $\mathsf{Vec}^\alpha G$ is required.

Theorem 3.5. *The monoidal categories* $\mathsf{Vec}^\alpha G$ *are pointed fusion categories.*

One method of describing monoidal equivalence classes is to first understand the possible functors that can occur between categories.

3.2 Functors between $\mathsf{Vec}^\alpha G$ and $\mathsf{Vec}^\beta H$

This section will focus on the proof of Theorem 3.6. This theorem is certainly known to experts in the field but I have been unable to find a citation with either a proof or a statement of it.

Theorem 3.6. *There is a bijection between the monoidal functors from* $\mathsf{Vec}^\alpha G \to \mathsf{Vec}^\beta H$ *up to monoidal natural isomorphism and the set*

$$\left\{(\theta, \Gamma) | \theta \in \mathsf{Hom}(G \to H), \Gamma \in H^2\left(G, \mathbb{K}^\times; \frac{\alpha}{\beta \cdot \theta}\right)\right\}.$$

The notation $\beta \cdot \theta$ refers to the function defined by $\beta \cdot \theta(g,h,k) = \beta(\theta(g), \theta(h), \theta(k))$ for $g, h, k \in G$. Additionally, $H^2\left(G, \mathbb{K}^\times; \frac{\alpha}{\beta\cdot\theta}\right)$ is the relative homology of G with respect to $\frac{\alpha}{\beta\cdot\theta}$. This is defined as $(\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)$ modulo the relation $x \sim y$ if $xy^{-1} \in \text{Im } \partial^2$.

The proof of this theorem will be the combination of the following four lemmas. The first two show that $H^2\left(G, \mathbb{K}^\times; \frac{\alpha}{\beta\cdot\theta}\right)$ is well defined and the second two are proving the bijection.

Lemma 3.7. *Given* β *a 3-cocycle on* H *and a group homomorphism* θ *from* G *to* H, $\beta \cdot \theta$ *is a 3-cocycle on* G.

Lemma 3.8. *Let* $\Phi \in (\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)$ *and* $\gamma \in \text{Im } \partial^2$. *Then* $\Phi\gamma \in (\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)$ *where* $\Phi\gamma$ *is the pointwise multiplication of* ϕ *and* γ.

Lemma 3.9. *There is a bijection between the monoidal functors from* $\mathsf{Vec}^\alpha G \to \mathsf{Vec}^\beta H$ *and the set*

$$\left\{ (\theta, \Gamma) | \theta \in \mathsf{Hom}(G \to H), \Gamma \in (\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)\right\}.$$

Denote $\mathcal{F}_{\theta,\Gamma}$ the monoidal natural transformation corresponding to a pair (θ, Γ) using the above bijection.

Lemma 3.10. *Two monoidal natural transformations* $\mathcal{F}_{\theta,\Gamma}$ *and* $\mathcal{F}_{\theta',\Gamma'}$ *are monoidally naturally isomorphic if and only if* $\theta = \theta'$ *and* $\Gamma(\Gamma')^{-1} \in \operatorname{Im} \partial^2$.

Proof of Lemma 3.7. Let β and θ be as given. Then for any four elements $g, h, k, l \in H$, β must satisfy

$$\beta(h, k, l)\beta(g, hk, l)\beta(g, h, k) = \beta(gh, k, l)\beta(g, h, kl).$$

Hence for any four elements $w, x, y, z \in G$, $\beta\cdot\theta$ satisfies

$$\begin{aligned}
&\beta\cdot\theta(x, y, z)\beta\cdot\theta(w, xy, z)\beta\cdot\theta(w, x, y)\\
&\qquad= \beta(\theta(x), \theta(y), \theta(z))\beta(\theta(w), \theta(x)\theta(y), \theta(z))\beta(\theta(w), \theta(x), \theta(y))\\
&\qquad= \beta(\theta(w)\theta(x), \theta(y), \theta(z))\beta(\theta(w), \theta(x), \theta(y)\theta(z))\\
&\qquad= \beta\cdot\theta(wx, y, z)\beta\cdot\theta(w, x, yz).
\end{aligned}$$

□

Proof of Lemma 3.8. Let $\Phi \in (\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)$ and $\gamma \in \operatorname{Im} \partial^2$. Then the proof follows the properties of the boundary maps in group cohomology. These boundary maps satisfy $\partial^3(fg) = \partial^3(f)\partial^3(g)$ for all $f, g \in C^2(G, \mathbb{K}^\times)$ and $\partial^3\cdot\partial^2(f) = 1$ for all $f \in C^1(G, \mathbb{K}^\times)$. Hence, as $\gamma \in \operatorname{Im} \partial^2$, $\gamma = \partial^2(f)$ for some f and so

$$\partial^3(\Phi\gamma) = \partial^3(\Phi)\partial^3(\partial^2(f)) = \frac{\alpha}{\beta\cdot\theta}.$$

□

This proves that $H^2\left(G, \mathbb{K}^\times; \frac{\alpha}{\beta\cdot\theta}\right)$ is indeed a well defined set. Therefore we can move on to proving the bijection.

Proof of Lemma 3.9. The outline for this proof is relatively simple. We will first show that given a monoidal functor $\mathcal{F}$, we can produce a pair $(\theta, \Gamma) \in (\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)$. Then it will be shown that given such a pair we can produce a monoidal functor.

Let V_g and W_h be the simple objects in $\mathsf{Vec}^\alpha G$ and $\mathsf{Vec}^\beta H$ respectively.

Initially let $\mathcal{F} = (F, \Phi, \phi)$ be a monoidal functor from $\mathsf{Vec}^\alpha G \to \mathsf{Vec}^\beta H$. Then ϕ is an isomorphism between W_e and $F(V_e)$. As the only element of $\mathsf{Vec}^\beta H$ isomorphic to W_e is W_e, this shows that $F(V_e) = W_e$. Therefore, as $\operatorname{End}(W_e) = \mathbb{K}$, ϕ corresponds to multiplication by some element in $\mathbb{K}^\times$. Observe that as $V_e = V_g \otimes V_{g^{-1}}$ there is an isomorphism

$$W_e \overset{\phi}{\simeq} F(V_e) = F(V_g \otimes V_{g^{-1}}) \overset{(\Phi_{g,g^{-1}})^{-1}}{\simeq} F(V_g) \otimes F(V_{g^{-1}}).$$

Thus $F(V_g)$ must be an invertible object in $\mathsf{Vec}^\beta H$. It follows from the definition of the tensor product that the only invertible objects in $\mathsf{Vec}^\beta H$ are the simple ones. Therefore there exists a map $\theta : G \to H$ such that $F(V_g) = V_{\theta(h)}$. Previously, it has been shown that $\theta(e) = e$ and $\theta(g^{-1}) = \theta(g)^{-1}$. Additionally the isomorphism

$$W_{\theta(gh)} = F(V_{gh}) = F(V_g \otimes V_h) \overset{(\Phi_{g,h})^{-1}}{\simeq} F(V_g) \otimes F(V_h) = V_{\theta(g)} \otimes V_{\theta(h)} = W_{\theta(g)\theta(h)}$$

shows that $\theta(gh) = \theta(g)\theta(h)$ and so θ is a group homomorphism from G to H. We have now produced the first half of the pair (θ, Γ). The second half will be found by closer analysis of the tensorator.

Consider Commutative Diagram 2.3 (The commutative diagram that relates the two possible maps from $(F(X) \cdot F(Y)) \cdot F(Z) \to F(X \otimes (Y \otimes Z)))$, with $X = V_g$, $Y = V_h$ and $Z = V_l$. Then, every object becomes $W_{\theta(ghl)}$ and so, as $\mathrm{End}(W_{\theta(ghl)}) = \mathbb{K}$, composition can be replaced by multiplication and the diagram exactly corresponds to the equation

$$\Phi_{g,h}\Phi_{gh,l}\alpha(g,h,l) = (\beta \cdot \theta)(g,h,l)\Phi_{h,l}\Phi_{g,hl}. \tag{3.3}$$

Therefore, defining Γ by $\Gamma(g,h) = \Phi_{g,h}$, a rearrangement of the above equation gives

$$\partial^3(\Gamma)(g,h,l) = \frac{\Gamma(h,l)\Gamma(g,hl)}{\Gamma(g,h)\Gamma(gh,l)} = \frac{\Phi_{h,l}\Phi_{g,hl}}{\Phi_{g,h}\Phi_{gh,l}} = \left(\frac{\alpha}{\beta \cdot \theta}\right)(g,h,l).$$

This shows that $\partial^3(\Gamma) = \frac{\alpha}{\beta \cdot \theta}$ and hence that $\Gamma \in (\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)$. This completes the proof of first direction of the bijection.

Next, let

$$(\theta, \Gamma) \in \left\{(\theta,\Gamma) | \theta \in \mathsf{Hom}(G \to H), \Gamma \in (\partial^3)^{-1}\left(\frac{\alpha}{\beta \cdot \theta}\right)\right\}$$

Then define a triple (F, Φ, ϕ) as follows. First F is defined to a functor from $\mathsf{Vec}^{\alpha}G$ to $\mathsf{Vec}^{\beta}H$ defined on a simple objects V_g by $F(V_g) = W_{\theta(g)}$ and extended by semisimplicity. Similarly, define Φ to be a natural isomorphism between $F(X) \otimes F(Y)$ and $F(X \otimes Y)$ with action $\Phi_{g,h}$ on a pair of simple objects V_g, V_h corresponding to multiplication by $\Gamma_{g,h}$ inside $\mathrm{End}(W_{gh}) \cong \mathbb{K}$. Finally, define ϕ to be an isomorphism from $F(V_e) = W_e$ to itself given by multiplication by $\Gamma_{e,e}^{-1} \in \mathrm{End}(W_e)$.

It follows easily from the definition of ∂^3 and Equation 3.3 that the triple (F, Φ, ϕ) satisfies the Diagram 2.3. Next consider Diagram 2.4 (The commutative diagram relating the left unitors). Plugging in the simple object $X = V_g$, this diagram corresponds to the equation in $W_{\theta(g)}$

$$\beta^{-1}(e,e,\theta(g)) = \Gamma^{-1}(e,e)\Gamma(e,g)\alpha^{-1}(e,e,g).$$

Rearranging, this is equivalent to requiring that

$$\left(\frac{\alpha}{\beta \cdot \theta}\right)(e,e,g) = \frac{\Gamma(e,g)}{\Gamma(e,e)}.$$

By the definition of Γ we find that

$$\begin{aligned}\left(\frac{\alpha}{\beta \cdot \theta}\right)(e,e,g) &= \partial^3\Gamma(e,e,g) \\ &= \frac{\Gamma(e,g)\Gamma(e,g)}{\Gamma(e,e)\Gamma(e,g)} \\ &= \frac{\Gamma(e,g)}{\Gamma(e,e)}.\end{aligned}$$

Therefore (F, Φ, ϕ) satisfies Diagram 2.4. A similar calculation shows that this triple also satisfies Diagram 2.5 (The commutative diagram relating the right unitors.) and so (F, Φ, ϕ) is indeed a monoidal functor from $\mathsf{Vec}^{\alpha}G$ to $\mathsf{Vec}^{\beta}H$. This completes the other direction of bijection and as such completes the proof of Lemma 3.9. □

This gives us a bijection between monoidal functors and the set $(\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)$. Next we show that the equivalence relation that two monoidal functors are equivalent if there is a monoidal natural isomorphism between them, will act thorough this bijection to quotient out $(\partial^3)^{-1}\left(\frac{\alpha}{\beta\cdot\theta}\right)$ by $\mathrm{Im}(\partial^2)$.

Proof of Lemma 3.10. Let $\mathcal{F}_{\theta,\Gamma} = (F, \Phi, \phi)$ and $\mathcal{F}_{\theta',\Gamma'} = (F', \Phi', \phi')$ be two monoidal functors from $\mathsf{Vec}^{\alpha}_{k_1} G \to \mathsf{Vec}^{\beta}_{k_2} H$.

Initially assume that $\theta \neq \theta'$. Then there exists exist some $g \in G$ such that $\theta(g) \neq \theta'(g)$. Hence, as the W_k are all simple objects, $\mathsf{Vec}^{\beta} H(W_{\theta(g)}, W_{\theta'(g)}) = 0$. Any natural isomorphism from $\mathcal{F}_{\theta,\Gamma}$ to $\mathcal{F}_{\theta',\Gamma'}$ needs to give an invertible map from $W_{\theta(g)} = F(V_g) \to F'(V_g) = W_{\theta'(g)}$. Clearly this is impossible and hence the two functors $\mathcal{F}$ and $\mathcal{F}'$ cannot be naturally isomorphic.

Therefore assume that $\theta = \theta'$ and now consider the conditions on Γ and Γ'.

Initially, assume that $\Gamma(\Gamma')^{-1}$ is in the image of ∂^2. Then there exists a $\gamma \in C^1(G, \mathbb{K}^\times)$ such that

$$\frac{\Gamma(g,h)}{\Gamma'(g,h)} = \frac{\gamma(g)\gamma(h)}{\gamma(gh)}. \tag{3.4}$$

Hence, define a natural isomorphism $\psi : F \to F'$ as follows. On a simple object V_g, ψ_g needs to be a map from $W_{\theta(g)} = F(V_g) \to F'(V_g) = W_{\theta(g)}$. Hence $\psi_g \in \mathrm{End}(W_{\theta(g)}) \cong \mathbb{K}$ and so define ψ_g to be multiplication by $\gamma(g)$. Extend this definition to all objects using semisimplicity. We claim that ψ is in fact a monoidal natural isomorphism from $\mathcal{F}_{\theta,\Gamma}$ to $\mathcal{F}_{\theta,\Gamma'}$.

First we need to prove that ψ is actually a natural isomorphism from F to F'. Considering only the simple objects, recall that there are no non-zero morphisms from $W_g \to W_h$ unless $g = h$. Hence given any morphism $f \in \mathsf{Vec}^{\alpha}_{k_1} G(V_g \to V_h)$, if $f = 0$ we trivially have that $\psi_g \circ 0 = 0 \circ \psi_g = 0$. Then, when $f \neq 0$, we must have that $g = h$ and so $f = \lambda 1_g$ and thus, again using the fact that $\mathrm{End}(W_{\theta(g)}) \cong \mathbb{K}$, $\psi_g \circ F(\lambda 1_g) = \psi_g \lambda = \lambda \psi_g = G(\lambda 1_g) \circ \psi_g$. Therefore ψ is indeed a natural transformation from $F \to F'$.

In order for ψ to be a monoidal natural transformation is needs to satisfy Commutative Diagrams 2.6 and 2.7 (these diagrams describe how monoidal natural transformations must interact with the actions of the tensorators and unit isomorphisms). Plugging $X = V_g$ and $Y = V_h$ into Diagram 2.6 yields the equality in $\mathrm{End}(W_{\theta(gh)})$

$$\gamma(g)\gamma(h)\Gamma'(g,h) = \Gamma(g,h)\gamma(gh). \tag{3.5}$$

Clearly this follows from Equation 3.4. Similarly, Diagram 2.7 produces an equality in $\mathrm{End}(W_e)$

$$\Gamma^{-1}_{e,e}\gamma(e) = \Gamma'^{-1}_{e,e}.$$

Once again this follows from Equation 3.4 by setting $g = h = e$. Therefore as required ψ is a monoidal natural isomorphism between $\mathcal{F}_{\theta,\Gamma} = (F, \Phi, \phi)$ and $\mathcal{F}_{\theta',\Gamma'} = (F', \Phi', \phi')$. This proves one direction of Lemma 3.10. In this case, the other direction is almost immediate.

Assume that there exists a monoidal natural isomorphism ψ from $\mathcal{F}_{\theta,\Gamma}$ to $\mathcal{F}_{\theta,\Gamma'}$. Define $\gamma(g)$ to be the element of $\mathbb{K} \cong \mathrm{End}(W_{\theta(g)})$ which corresponds to the map ψ_g. Then, the commutative diagram 2.6 with $X = V_g$ and $Y = V_h$ will again produce Equation 3.5 which is clearly equivalent to Equation 3.4 which shows that $\Gamma(\Gamma')^{-1} \in \mathrm{Im}\,\partial^1$. □

Combining the four aforementioned lemmas proves Theorem 3.6.

This bijection has a nice interaction with composition of functors. Given two monoidal natural transformations $\mathcal{F}_{\theta_1,\Gamma_1} : \mathsf{Vec}^{\alpha}_{k_1} G \to \mathsf{Vec}^{\beta}_{k_2} H$ and $\mathcal{F}_{\theta_2,\Gamma_2} : \mathsf{Vec}^{\beta}_{k_2} H \to \mathsf{Vec}^{\gamma}_{k_3} K$, their composition is given by $\mathcal{F}_{\theta_2,\Gamma_2} \circ \mathcal{F}_{\theta_1,\Gamma_1} = \mathcal{F}_{\theta_2\circ\theta_1,\Gamma_1(\Gamma_2\cdot\theta_1)}$. In particular, observe that if θ is invertible, then $\mathcal{F}_{\theta,\Gamma}$

is also invertible with two-sided inverse $\mathcal{F}_{\theta^{-1},\Gamma^{-1}.\theta^{-1}}$. On the other hand, if θ is not invertible then $\mathcal{F}_{\theta,\Gamma}$ cannot have an two-sided inverse[2].

This leads to the following useful corollary. While the version presented and proved here only contains an if statement, it turns out that this will still be true if the if statement is replaced by an if and only if [1, 16].

Corollary 3.11. *Two categories* $\mathsf{Vec}^{\alpha}G$ *and* $\mathsf{Vec}^{\beta}H$ *are monoidally equivalent if there exists an invertible* $\theta \in \mathsf{Hom}(G \to H)$ *such that* $\frac{\alpha}{\beta\cdot\theta} \in \operatorname{Im}\partial^3$.

Proof. The proof of the corollary is almost immediate from the previous discussion. If there exists an invertible $\theta \in G$ such that $\frac{\alpha}{\beta\cdot\theta} \in \operatorname{Im}\partial^3$ then by definition there exists a Γ such that $\partial^3\Gamma = \frac{\alpha}{\beta\cdot\theta}$. Therefore $\mathcal{F}_{\theta,\Gamma}$ is an invertible monoidal functor from $\mathsf{Vec}^{\alpha}G \to \mathsf{Vec}^{\beta}H$ and so these categories are monoidally equivalent. □

3.3 Equivalence classes of Pointed Fusion Categories

Corollary 3.11 is a powerful tool for establishing equivalences of pointed fusion categories.

Observe that given any pair α, α' of cohomologous 3-cocycles, Corollary 3.11 with $\theta = 1_G$ shows that $\mathsf{Vec}^{\alpha}G$ and $\mathsf{Vec}^{\alpha'}G$ must be monoidally equivalent. Combining this with Theorem 2.29 allows us to conclude that every category of the form $\mathsf{Vec}^{\alpha}G$ is equivalent to some category $\mathsf{Vec}^{\beta}G$ where β is unitary. This is a useful observation because, for unitary cocycles, both the unitors of the corresponding category become trivial. With this observation in mind, let us jump back and prove that all categories $\mathsf{Vec}^{\alpha}G$ are fusion categories.

Proof of Theorem 3.5. It is well known that equivalences of categories preserve properties. Therefore it is good enough to prove this Theorem in the case that that α is unitary.

Previously it has been shown that $\mathsf{Vec}^{\alpha}G$ is a semisimple linear monoidal category, with finitely many simple objects $\{\delta_g\}_{g\in G}$ and a simple unit object δ_e. We must show then that $\mathsf{Vec}^{\alpha}G$ is pointed and rigid.

Initially observe that given any simple object δ_g, we have the equality

$$\delta_g \otimes \delta_{g^{-1}} = \delta_{g^{-1}} \otimes \delta_g = \delta_e.$$

Therefore $\mathsf{Vec}^{\alpha}G$ is a pointed category.

Next, let V be any object in $\mathsf{Vec}^{\alpha}G$. Define its dual to be

$$V^* = \bigoplus_{g\in G} (V_{g^{-1}})^*$$

where $(V_{g^{-1}})^*$ is the usual vector space dual of $V_{g^{-1}}$. Then we need an evaluation map $\epsilon_V : V^* \otimes V \to I$ and a co-evaluation map $\eta_V : I \to V \otimes V^*$. Let $\{v_{g_i}\}$ be a basis for V_g with $v^*_{g_i}$ the corresponding dual basis of $V^*_{g^{-1}}$. Then define for any $k \in \delta_e \cong \mathbb{K}$

$$\eta_V(k) = k \sum_{g\in G} \omega^{-1}(g, g^{-1}, g) \sum_i v_{g_i} \otimes v^*_{g_i}.$$

Similarly to in Vec, ϵ_V will be exactly given by function application. The two equalities that this is required to satisfy follow either instantly, or from the observation that the 3-cocycle condition on the tuple (g, g^{-1}, g, g^{-1}) implies that $\omega(g, g^{-1}, g) = \omega^{-1}(g^{-1}, g, g^{-1})$. Left duals will be defined in an identical manner. □

[2]It could have a 1-sided inverse however.

Additionally, we can also give $\mathsf{Vec}^{\alpha}G$ a pivotal structure. This is defined on the simple objects by $\phi_g = \omega(g, g^{-1}, g)\overline{\phi}$ where $\overline{\phi}$ is the usual pivotal structure on Vec. With this pivotal structure it can be easily checked that $\mathsf{Vec}^{\alpha}G$ is spherical. Note that inside any G graded piece, the constants from ϕ_V and η_V or ϕ_V^{-1} and η_{V^*} cancel out and so the trace map is identical to Vec. Hence as all morphisms are G-graded and Vec is spherical, $\mathsf{Vec}^{\alpha}G$ must also be spherical. As might be expected, with these structures defined, the dimension of any object is exactly the dimension of the vector space.

Looking again at Corollary 3.11, observe that fixing any 3-cocycle α, every category $\mathsf{Vec}^{\alpha\cdot\theta}G$ for $\theta \in \mathrm{Aut}(G)$ is equivalent. This allows for a stronger statement of 3.1.

Fix a positive integer k and let S_k denote the set of all isomorphism classes of finite groups of order k. For each equivalence class in S choose a representative $G_{k,i}$ and define the set $\{H_{k,i,j}\}$ to be the set of orbits of $H^3(G_{k,i}, \mathscr{K}^{\times})$ under the action of $\mathrm{Aut}(G_{k,i})$ with $\theta \in \mathrm{Aut}(G_{k,i})$ acting on a 3-cocycle cohomology class with representative α by the map

$$\theta \cdot [\alpha] = [\alpha \cdot \theta^{-1}].$$

Let $\beta_{k,i,j}$ be a set of representative cocycles of a representative cohomology class of $H_{k,i,j}$. By Theorem 2.29, it can be assumed that β is unitary.

Theorem 3.12. *Let $\mathscr{C}$ be a pointed fusion category of rank k. Then there exists a $G_{k,i}$ and $\beta_{k,i,j}$ such that $\mathscr{C}$ is equivalent to $\mathsf{Vec}^{\beta_{k,i,j}}G_{k,i}$.*

Proof. By Theorem 3.1, $\mathscr{C}$ is equivalent $\mathsf{Vec}^{\alpha}G$ for some G, α. Then as $k = \mathrm{rank}(\mathscr{C}) = \mathrm{rank}(\mathsf{Vec}^{\alpha}G) = |G|$, G must be isomorphic to $G_{k,i}$ for some i. Letting θ be that isomorphism, Corollary 3.11 shows that $\mathsf{Vec}^{\alpha}G$ is equivalent to $\mathsf{Vec}^{\alpha\circ\theta^{-1}}G_{k,i}$. Then, $[\alpha \circ \theta^{-1}]$ is in the same orbit at $\beta_{k,i,j}$ for some j and so $\mathsf{Vec}^{\alpha\circ\theta^{-1}}G_{k,i}$ is equivalent to $\mathsf{Vec}^{\beta_{k,i,j}}G_{k,i}$. □

This theorem provides the starting point for the construction of a database of modular data of Drinfeld centres of pointed fusion categories. Due to this quotient by the automorphism group, the number of non equivalent fusion categories can be massively reduced from uncomputable values down to more reasonable values. This is particularly apparent when dealing with groups with large automorphism groups such as powers of $\mathbb{Z}/2\mathbb{Z}$. While $\left|H^3((\mathbb{Z}/2\mathbb{Z})^5, \mathbb{C}^{\times})\right| = 2^{25} = 33554432$, the number of orbits under the action of the automorphism group is merely 88.

Theorem 3.12 is a special case of a much more general theorem proven by an ANU PhD student, Cain Edie-Michell, in his soon to be published paper [4]. Broadly, given any monoidal category $\mathscr{C}$ we can define a group extension $\mathscr{D}$ of $\mathscr{C}$. As a category, $\mathscr{D}$ is a direct sum of copies of $\mathscr{C}$, each sitting over a difference element of G

$$\mathscr{D} = \bigoplus_{g\in G} \mathscr{C}_g.$$

Similarly to above, we require that the tensor product interact with this decomposition in the sense that

$$\otimes : \mathscr{C}_g \times \mathscr{C}_h \to \mathscr{C}_{gh}.$$

Additionally, when we consider the sub monoidal category $\mathscr{C}_e$ with monoidal structure given by the restriction from $\mathscr{D}$, this is required equivalent to be the original category.

Again there are a variety of choices that can be made in defining the monoidal structure. Edie-Michell gives a complete description of the equivalence classes of the possible monoidal structures on these categories. Corollary 3.11 is exactly the specialisation of Edie-Michell's more general theorem to the case where $\mathscr{C} = \mathsf{Vec}$. This being said, the method of proof is different.

Edie-Michell does not produce a more general result of Theorem 3.6 and then use this to classify the equivalences like was done here. It is likely however that the techniques he uses could be retooled to expand Theorem 3.6.

Chapter 4

Modular Data for the Drinfeld Centre of Twisted G-graded Vector Spaces

For simplicity, in this chapter and for the rest of this thesis all fields will be assumed to be $\mathbb{C}$. Most of the results here hold true over arbitrary fields but a couple of them do require the field to be algebraically complete and or to have characteristic[1] 0.

The goal of this section is to compute the S and T matrices for $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$. This will be done by constructing a quasitriangular quasi Hopf Algebra $D^{\omega}G$ and showing that $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop} \cong \mathrm{Rep}(D^{\omega}G)$. This will then allow us to translate the morphisms defining the S and T matrices into algebraic expressions in $D^{\omega}G$ which will then allow us to simplify and solve them. The main result from this section is the following theorem. For a group G and a unitary 3-cocycle ω we define

$$\theta_g(x,y) = \frac{\omega(g,x,y)\omega(x,y,(xy)^{-1}gxy)}{\omega(x,x^{-1}gx,y)}. \tag{4.1}$$

Additionally, for each conjugacy class K, fix a representative a and for every other $g \in K$ a group element a_g such that $a = a_g g a_g^{-1}$ and recall the definition of the centraliser of a, $C(a) = \{g \in G | ag = ga\}$.

Theorem 4.1. *The simple objects of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ correspond to tuples (K, a, ρ_a) with K a conjugacy class of G with representative a and ρ_a an irreducible θ_a-projective representation of the centraliser of a, $C(a)$. Then the entries of the S and T matrices of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ are given by the formulas*

$$T_{(K,a,\rho_a),(L,b,\psi_b)} = \delta_{K,L}\delta_{\rho_a,\psi_b}\frac{\chi_{\rho_a}(a)}{\chi_{\rho_a}(e)}. \tag{4.2}$$

and

$$S_{(K,a,\rho_a),(L,b,\psi_b)} = \frac{1}{|G|}\sum_{\substack{g\in K\\ h\in L\cap C(g)}}\left(\frac{\theta_a(a_g,h)\theta_a(a_g h, a_g^{-1})\theta_b(b_h,g)\theta_b(b_h g, b_h^{-1})}{\theta_g(a_g^{-1},a_g)\theta_h(b_h^{-1},b_h)}\right)^* \times \chi^*_{\rho_a}(a_g h a_g^{-1})\chi^*_{\psi_b}(b_h g b_h^{-1}). \tag{4.3}$$

These equations for the S and T matrices have appeared in the literature before such as Equations 5.23 and 5.24 in a paper by Antoine Coste, Terry Gannon and Philippe Ruelle [5]. Simplified versions of these formula when ω equals 1 have appeared throughout the literature

[1] More explicitly often the characteristic is required to not divide some integer, (usually the order of a group) and it is easier to just work over $\mathbb{C}$ than explicitly deal with this each time.

[5, 17–19]. This being said, no detailed derivation of these formulas has appeared previously.

4.1 An Intricate Hopf Algebra

Let G be a finite group with identity e and $\omega \in H^3(G, \mathbb{C})$ a unitary 3-cocycle. Let

$$\gamma_x(h, l) = \frac{\omega(h, l, x)\omega(x, x^{-1}hx, x^{-1}lx)}{\omega(h, x, x^{-1}lx)}. \tag{4.4}$$

and recall the definition of $\theta_g(x, y)$ given in Equation 4.1.

Then the quasitriangular quasi Hopf algebra $D^\omega G$ is defined following [14]. Start with the vector space over $\mathbb{C}$ generated by symbols $\delta_g\overline{x}$ with $g, x \in G$. The algebra structure is

$$\nabla(\delta_g\overline{x}, \delta_h\overline{y}) = \theta_g(x, y)\delta_{g,xhx^{-1}}\delta_g\overline{xy} = \begin{cases} \theta_g(x, y)\delta_g\overline{xy} & \text{if } g = xhx^{-1} \\ 0 & \text{else} \end{cases} \tag{4.5}$$

$$\eta(k) = k \sum_{g \in G} \delta_g\overline{e}. \tag{4.6}$$

where $\delta_{g,h}$ is the Kronecker delta function. The coalgebra structure is

$$\Delta(\delta_g\overline{x}) = \sum_{h \in G} \gamma_x(h, h^{-1}g)\delta_h\overline{x} \otimes \delta_{h^{-1}g}\overline{x} \tag{4.7}$$

$$\epsilon(\delta_g\overline{x}) = \delta_{g,e}. \tag{4.8}$$

This structure is not coassociative but is quasi coassociative with invertible element

$$\Phi = \sum_{g,h,k \in G} \omega(g, h, k)\delta_g\overline{e} \otimes \delta_h\overline{e} \otimes \delta_k\overline{e}.$$

This means that

$$(\Delta \otimes 1) \circ \Delta(h) = \Phi(1 \otimes \Delta) \circ \Delta(h)\Phi^{-1}.$$

Next, the quasi Hopf algebra structures are

$$S(\delta_h\overline{g}) = \theta_{h^{-1}}(g, g^{-1})^{-1}\gamma_g(h, h^{-1})^{-1}\delta_{g^{-1}h^{-1}g}\overline{g^{-1}} \tag{4.9}$$

with $\alpha = \mathbf{1} = \eta(1) = \sum_{g \in G} \delta_g\overline{e}$ and $\beta = \sum_g \omega(g, g^{-1}, g)\delta_g\overline{e}$. Additionally, the quasitriangular element is

$$R = \sum_{g,h \in G} \delta_g\overline{e} \otimes \delta_h\overline{g}.$$

There is clearly a lot that would need to be checked here to show that these definitions indeed produce a quasitriangular quasi Hopf algebra. For brevity most of these will be skipped but they all essentially follow from repeated use of the 3-cocycle condition for ω as will be seen in the following proof.

Proof that $(\Delta \otimes 1) \circ \Delta(h)\Phi = \Phi(\Delta \otimes 1) \circ \Delta(h)$ *for all* $h \in D^\omega G$. As both Δ and multiplication by Φ are linear maps, it suffices to show that this equality holds on the vector space generators of $D^\omega G$. Therefore, observe that on an arbitrary generator $(\delta_g\overline{x})$, the left hand side expands to

$$(\Delta \otimes 1) \circ \Delta(\delta_g\overline{x})\Phi = (\Delta \otimes 1)\Big(\sum_{h \in G} \gamma_x(h, h^{-1}g)\delta_h\overline{x} \otimes \delta_{h^{-1}g}\overline{x}\Big)\Phi$$

$$= \sum_{h,k\in G} \gamma_x(h,h^{-1}g)\gamma_x(k,k^{-1}h)\omega(x^{-1}kx,x^{-1}k^{-1}hx,x^{-1}h^{-1}gx)\delta_k\overline{x}\otimes\delta_{k^{-1}h}\overline{x}\otimes\delta_{h^{-1}g}\overline{x},$$

whereas the right hand side expands to

$$\begin{aligned}\Phi(1\otimes\Delta)\circ\Delta(\delta_g\overline{x}) &= \Phi(1\otimes\Delta)\Big(\sum_{h\in G}\gamma_x(h,h^{-1}g)\delta_h\overline{x}\otimes\delta_{h^{-1}g}\overline{x}\Big)\Phi^{-1}\\ &= \sum_{h,k\in G}\omega(h,k,k^{-1}h^{-1}g)\gamma_x(h,h^{-1}g)\gamma_x(k,k^{-1}h^{-1}g)\delta_h\overline{x}\otimes\delta_k\overline{x}\otimes\delta_{k^{-1}h^{-1}g}\overline{x}\\ &= \sum_{h,k\in G}\omega(h,h^{-1}k,k^{-1}g)\gamma_x(h,h^{-1}g)\gamma_x(h^{-1}k,k^{-1}g)\delta_h\overline{x}\otimes\delta_{h^{-1}k}\overline{x}\otimes\delta_{k^{-1}g}\overline{x}\\ &= \sum_{h,k\in G}\omega(k,k^{-1}h,h^{-1}g)\gamma_x(k,k^{-1}g)\gamma_x(k^{-1}h,h^{-1}g)\delta_k\overline{x}\otimes\delta_{k^{-1}h}\overline{x}\otimes\delta_{h^{-1}g}\overline{x}.\end{aligned}$$

In the last two steps k was relabelled by $h^{-1}k$ and then the labels h and k were swapped. Clearly, these two expressions will be equal in general if

$$\omega(k,k^{-1}h,h^{-1}g)\gamma_x(k,k^{-1}g)\gamma_x(k^{-1}h,h^{-1}g) = \omega(x^{-1}kx,x^{-1}k^{-1}hx,x^{-1}h^{-1}gx)\gamma_x(h,h^{-1}g)\gamma_x(k,k^{-1}h).$$

In order to show this equality the first step will be to expand out all the γ's. Additionally, to simplify the expressions replace $k^{-1}h$ and $h^{-1}g$ with l and m respectively. Then, the left hand side of the equation becomes

$$LHS = \frac{\omega(k,l,m)\omega(k,lm,x)\omega(x,x^{-1}kx,x^{-1}lmx)\omega(l,m,x)\omega(x,x^{-1}lx,x^{-1}mx)}{\omega(k,x,x^{-1}lmx)\omega(l,x,x^{-1}mx)}$$

and the right hand side becomes

$$RHS = \frac{\omega(x^{-1}kx,x^{-1}lx,x^{-1}mx)\omega(kl,m,x)\omega(x,x^{-1}klx,x^{-1}mx)\omega(k,l,x)\omega(x,x^{-1}kx,x^{-1}lx)}{\omega(kl,x,x^{-1}mx)\omega(k,x,x^{-1}lx)}.$$

Using the 3-cocycle condition on the tuple (k,l,m,x) on the left hand side simplifies it to

$$LHS = \frac{\omega(k,l,mx)\omega(x,x^{-1}lx,x^{-1}mx)\omega(kl,m,x)\omega(x,x^{-1}kx,x^{-1}lmx)}{\omega(k,x,x^{-1}lmx)\omega(l,x,x^{-1}mx)}.$$

In a similar vein, the 3-cocycle on $(x,x^{-1}kx,x^{-1}lx,x^{-1}mx)$ simplifies the right hand side to

$$RHS = \frac{\omega(kx,x^{-1}lx,x^{-1}mx)\omega(k,l,x)\omega(kl,m,x)\omega(x,x^{-1}kx,x^{-1}lmx)}{\omega(kl,x,x^{-1}mx)\omega(k,x,x^{-1}lx)}.$$

Definitionally, these expressions will be equal if and only if 1 is equal to

$$\frac{LHS}{RHS} = \frac{\omega(k,l,mx)\omega(x,x^{-1}lx,x^{-1}mx)}{\omega(k,x,x^{-1}lmx)\omega(l,x,x^{-1}mx)}\frac{\omega(kl,x,x^{-1}mx)\omega(k,x,x^{-1}lx)}{\omega(kx,x^{-1}lx,x^{-1}mx)\omega(k,l,x)}.$$

Using the 3-cocycle condition on $(k,l,x,x^{-1}mx)$ simplifies this expression further to

$$\frac{LHS}{RHS} = \frac{\omega(k,lx,x^{-1}mx)\omega(x,x^{-1}lx,x^{-1}mx)\omega(k,x,x^{-1}lx)}{\omega(k,x,x^{-1}lmx)\omega(kx,x^{-1}lx,x^{-1}mx)}.$$

Finally, the 3-cocycle condition on $(k,x,x^{-1}lx,x^{-1}mx)$ shows that this expression is equal to 1 and so, as all simplifying steps were equivalences, this proves as required that for all $h \in D^\omega G$

$$(\Delta \otimes 1) \circ \Delta(h)\Phi = \Phi(\Delta \otimes 1) \circ \Delta(h). \qquad \square$$

More generally, all the equalities that need to be satisfied follow from either a single application of the 3-cocycle condition, a consequence of ω being unitary and hence normalized, or one of the following three identities which can be themselves proven by repeated use of the 3-cocycle condition.

$$\theta_g(x,y)\theta_g(xy,z) = \theta_g(x,yz)\theta_{x^{-1}gx}(y,z) \tag{4.10}$$

$$\theta_g(x,y)\theta_h(x,y)\gamma_x(g,h)\gamma_y(x^{-1}gx, x^{-1}hx) = \theta_{gh}(x,y)\gamma_{xy}(g,h) \tag{4.11}$$

$$\gamma_x(g,h)\gamma_x(gh,k)\omega(x^{-1}gx, x^{-1}hx, x^{-1}kx) = \gamma_x(h,k)\gamma_x(g,hk)\omega(g,h,k). \tag{4.12}$$

Observe that by renaming $k \to g, k^{-1}h \to h, h^{-1}g \to k$, Equation 4.12 was just shown to hold in the preceding proof. One important observation from Equation 4.10 is that for any element $g \in G$, $\theta_g(x,y)$ is a 2-cocycle on $C(g)$.

4.2 The Modular Equivalence between $\mathrm{Rep}(D^\omega G)$ and $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}})^{bop}$

The main goal of this section will be to prove the following theorem.

Theorem 4.2. *The ribbon fusion categories* $\mathrm{Rep}(D^\omega G)$ *and* $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ *are equivalent.*

Note the $^{-1}$ and bop in $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$. These are unfortunate consequence of differing conventions in the algebraic and categorical worlds with the associator on $\mathrm{Rep}(D^\omega G)$ originating from Φ^{-1} as opposed to Φ and the half braidings in $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ being required to satisfy a right group action as opposed to a left group action.

This theorem will be proven incrementally, by use of the following lemmas.

Lemma 4.3. *There is an equivalence on the level of categories between* $\mathrm{Rep}(D^\omega G)$ *and* $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$.

Lemma 4.4. *The equivalence on the level of categories between* $\mathrm{Rep}(D^\omega G)$ *and* $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ *extends to a monoidal equivalence.*

Lemma 4.5. *The monoidal equivalence between* $\mathrm{Rep}(D^\omega G)$ *and* $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ *can be made into a braided equivalence.*

Lemma 4.6. *The monoidal equivalence between* $\mathrm{Rep}(D^\omega G)$ *and* $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}}G\right)^{bop}$ *can be made into a pivotal equivalence.*

Recall from Section 2.2.3 that a representation of $D^\omega G$ is exactly a left module of the underlying algebra. Looking back at Equation 4.5, observe that $\nabla(\delta_g\overline{x}, \delta_h\overline{y}) \neq 0$ if and only if $g = xhx^{-1}$. In particular, this means that g and h must be in the same conjugacy class. Hence the underlying algebra of $D^\omega G$ can be written as the direct sum over the conjugacy classes $I(G)$ of G

$$D^\omega G = \bigoplus_{K \in I(G)} D^\omega(K,G).$$

Here $D^\omega(K,G)$ is the subalgebra of $D^\omega G$ generated by $\delta_g\overline{x}$ for $g \in K$ and $x \in G$ with identity $\sum_{g\in K} \delta_g\overline{e}$. This splitting of $D^\omega G$ will be essential in describing the simple objects of $\mathrm{Rep}(D^\omega G)$.

The functor $Eq : \mathrm{Rep}(D^\omega G) \to \mathcal{Z}\big(\mathsf{Vec}^{\omega^{-1}} G\big)^{bop}$ is defined as follows. Let (V, ρ) be a left $D^\omega G$ module. For any $g \in G$, as $\nabla(\delta_g \bar{e}, \delta_g \bar{e}) = \delta_g \bar{e}$, $\rho(\delta_g \bar{e})$ must be a projection matrix and thus diagonalizable. Additionally, for any other $h \in G$,

$$\nabla(\delta_g \bar{e}, \delta_h \bar{e}) = \nabla(\delta_h \bar{e}, \delta_g \bar{e}) = \begin{cases} \delta_g \bar{e} & \text{if } g = h \\ 0 & \text{otherwise.} \end{cases}$$

Therefore the set of matrices $S = \{\rho(\delta_g \bar{e})\}_{g \in G}$ must be simultaneously diagonalisable as they all commute. Hence, with respect to a basis $\{v_1, \cdots, v_n\}$ diagonalizing S, $\rho(\delta_g \bar{e})$ will be a projection onto some subspace $\{v_{g_1}, \cdots, v_{g_{k_g}}\}$. As $\sum_{g \in G} \delta_g \bar{e} = \mathbf{1}$, every basis element will be in the image of a unique $\rho(\delta_g \bar{e})$ and so V splits as

$$V = \bigoplus_{g \in G} V_g.$$

where $V_g = \mathrm{Im}(\rho(\delta_g \bar{e})) = \mathrm{Span}(v_{g_1}, \cdots, v_{g_{k_g}})$.

Two useful observation are as follows. Given any element $\delta_g \overline{x}$, note the equality

$$\rho(\delta_g \overline{x}) \circ \rho(\delta_{x^{-1}gx} \bar{e}) = \rho\big(\nabla(\delta_g \overline{x}, \delta_{x^{-1}gx} \bar{e})\big) = \rho(\delta_g \overline{x}) = \rho\big(\nabla(\delta_g \bar{e}, \delta_g \overline{x})\big) = \rho(\delta_g \bar{e}) \circ \rho(\delta_g \overline{x})$$

As $\rho(\delta_{x^{-1}gx} \bar{e})$ and $\rho(\delta_g \bar{e})$ are projections onto V_g and $V_{x^{-1}gx}$, this equality shows that $\rho(\delta_g \overline{x})$ is a linear map from $V_{x^{-1}gx} \to V_g$ and acts as 0 on V_h for $h \neq x^{-1}gx$. As $\delta_g \overline{x}$ is invertible, V_g and $V_{x^{-1}gx}$ must have the same dimension for all $x \in G$. Hence for every conjugacy class K, the dimension of V_g is fixed for all $g \in K$.

Define $D^\omega(g, C(g))$ to be the subalgebra of $D^\omega(K, G)$ generated by $\delta_g \overline{x}$ for $x \in C(g)$. Then, V_g is a left module of $D^\omega(g, C(g))$ with action given by the restricting the domain of ρ to $D^\omega(g, C(g))$ and V_g.

So far we have shown that a left $D^\omega G$ module is naturally a G-graded vector space however, elements of $\mathcal{Z}\big(\mathsf{Vec}^{\omega^{-1}} G\big)^{bop}$ also require a half braiding, $\overline{\beta}$. Let W_g be the one dimensional G-graded vector space sitting over the element $g \in G$ and let $\beta_g = \rho\left(\sum_{k \in G} \delta_k \overline{g}\right)$. Define a map $\overline{\beta}_g$ from $V \otimes W_g \to W_g \otimes V$ by $\overline{\beta}_g = F \circ (\beta_g^{-1} \otimes 1)$. With F, the flip map defined in Chapter 2.2.

Lemma 4.7. *This map $\overline{\beta}_g$ defines a half braiding on the simple objects of* $\mathsf{Vec}^{\omega^{-1}} G$ *and as such extends to a half braiding on all of* $\mathsf{Vec}^{\omega^{-1}} G$.

Proof. First we show that β_g^{-1} exists and $\overline{\beta}_g$ is a G-graded map. Then we will show that $\overline{\beta}_g$ is natural and that it satisfies the half braiding equality $(\omega^{-1}_{g,h,V})^{-1} \circ (1 \otimes \overline{\beta}_h) \circ \omega^{-1}_{g,V,h} \circ (\overline{\beta}_g \otimes 1) \circ (\omega^{-1}_{V,g,h})^{-1} = \overline{\beta}_{gh}$.

Initially, observe that $\left(\sum_{k \in G} \delta_k \overline{g}\right)^{-1} = \sum_{l \in G} \theta_{glg^{-1}}(g, g^{-1})^{-1} \delta_l \overline{g^{-1}}$ and so β_g^{-1} exists. Next note that $(\beta_g)^{-1}$ restricted to $V_{hg^{-1}}$ will be a map from $V_{hg^{-1}}$ to $\to V_{g^{-1}h}$. Hence as

$$V \otimes W_g = \bigoplus_{h \in G} V_{hg^{-1}}$$

and

$$W_g \otimes V = \bigoplus_{h \in G} V_{g^{-1}h},$$

$\overline{\beta}_g$ is indeed a G-graded map.

Naturality of $\overline{\beta}_g$ is immediate as it is acts by the identity map on W_g. Finally, we are left with showing the half braiding condition. To make the calculations simpler, let us first invert both sides of the equality to get

$$\omega^{-1}_{V,g,h} \circ (\overline{\beta}_g^{-1} \otimes 1) \circ (\omega^{-1}_{g,V,h})^{-1} \circ (1 \otimes \overline{\beta}_h^{-1}) \circ \omega^{-1}_{g,h,V} = \overline{\beta}_{gh}^{-1}.$$

Then, pick a representative element $(u, w, v) \in W_g \otimes W_h \otimes V$ with $v = \sum_{k \in G} v_k$ where each $v_k \in V_k$. The right hand side of the proposed equivalence acts upon this element by

$$u \otimes w \otimes v \xmapsto{\overline{\beta}_{gh}^{-1}} \beta_{gh}(v) \otimes u \otimes w = \sum_{k \in G} \beta_{gh}(v_k) \otimes u \otimes w.$$

In order to calculate the action of the left hand side, observe that if $v_k \in V_k$, then $\beta_h(v_k) \in V_{hkh^{-1}}$. Using this fact we can derive that left hand side will act by

$$\begin{aligned}
u \otimes w \otimes v &= \sum_k u \otimes w \otimes v_k \\
&\xmapsto{\omega^{-1}_{g,h,V}} \sum_k \omega^{-1}(g,h,k) u \otimes w \otimes v_k \\
&\xmapsto{1 \otimes \overline{\beta}_h^{-1}} \sum_k \omega^{-1}(g,h,k) u \otimes \beta_h(v_k) \otimes w \\
&\xmapsto{(\omega^{-1}_{g,V,h})^{-1}} \sum_k \frac{\omega^{-1}(g,h,k)}{\omega^{-1}(g,hkh^{-1},h)} u \otimes \beta_h(v_k) \otimes w \\
&\xmapsto{\beta_g^{-1} \otimes 1} \sum_k \frac{\omega^{-1}(g,h,k)}{\omega^{-1}(g,hkh^{-1},h)} \beta_g(\beta_h(v_k)) \otimes u \otimes w \\
&\xmapsto{\omega^{-1}_{V,g,h}} \sum_k \frac{\omega^{-1}(g,h,k)\omega^{-1}(ghk(gh)^{-1},g,h)}{\omega^{-1}(g,hkh^{-1},h)} \beta_g(\beta_h(v_k)) \otimes u \otimes w \\
&= \sum_k \theta^{-1}_{ghk(gh)^{-1}}(g,h) \beta_g(\beta_h(v_k)) \otimes u \otimes w.
\end{aligned}$$

Simply observe now that

$$\begin{aligned}
\beta_g(\beta_h(v_k)) &= \rho\Big(\Big(\sum_{l,m} \nabla(\delta_l \overline{g}, \delta_m \overline{h}\Big), v_k\Big) \\
&= \rho\Big(\sum_l \theta_l(g,h) \delta_l \overline{gh}, v_k\Big) \\
&= \theta_{ghk(gh)^{-1}} \rho(\delta_{ghk(gh)^{-1}} \overline{gh}, v_k) \\
&= \theta_{ghk(gh)^{-1}} \beta_{gh}(v_k).
\end{aligned}$$

Substituting this in shows that the left hand side and the right hand side act identically and so the proposed morphism is indeed a half braiding. □

This defines how Eq maps objects in $D^\omega G$ to objects in $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$. On, morphisms, Eq will correspond to the identity map as a morphism that commutes with the action of $D^\omega G$ must commute with $\rho(\delta_g \overline{e})$ and β_g which means it must respect the derived G-grading and will commute with the half braiding.

It also needs to be shown that Eq has an inverse[2]. For brevity I will only give a brief outline of this inverse as explaining the finer details is essentially a rehash of the work done above.

[2]In general an equivalence of categories requires a slightly weaker condition than in inverse existing but in this case the constructed functor Eq is really on the nose invertible.

Given, an object $(V, \overline{\beta})$ in $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$, the corresponding $D^\omega G$ module is created by forgetting the G grading on V and equipping it with the action

$$\rho(\delta_g \overline{x}) = (\overline{\beta}_x^{-1}) \circ P(x^{-1} g x)$$

where $P(h)$ is the projection of V onto V_h. Morphisms will again be mapped across by the identity map.

This completes the proof of Lemma 4.3.

Next we move on to showing that this equivalence between Rep $\left(D^\omega G\right)$ and $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$ is a monoidal one.

Proof of Lemma 4.4. Let (V, ρ) and (W, ψ) be two left $D^\omega G$ modules. Denote the associated elements of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$ by $Eq(V, \rho) = (V, \overline{\alpha})$ and $Eq(W, \psi) = (W, \overline{\beta})$. Initially let us show that

$$Eq(V, \rho) \otimes Eq(W, \psi) = Eq((V, \rho) \otimes (W, \psi)) = Eq(V \otimes W, (\rho \otimes \psi) \circ \Delta).$$

Recall that

$$\left(V \otimes W\right)_g = \bigoplus_{hk=g} V_h \otimes W_k = \bigoplus_{h \in G} V_h \otimes W_{h^{-1}g}$$

and V_g is exactly the fixed subspace of V corresponding to the projection by the action of $\delta_g \overline{e}$. Then the action on $V \otimes W$ of $\delta_g \overline{e}$ is given by

$$(\rho \otimes \psi)\big(\Delta(\delta_g \overline{e})\big) = (\rho \otimes \psi)\Big(\sum_{h \in G} \delta_h \overline{e} \otimes \delta_{h^{-1}g} \overline{e}\Big) = \Big(\sum_{h \in G} \rho(\delta_h \overline{e}) \otimes \psi(\delta_{h^{-1}g} \overline{e})\Big).$$

Hence as required, $(\rho \otimes \psi)\big(\Delta(\delta_g \overline{e})\big)$ is exactly a projection onto the space

$$\bigoplus_{h \in G} V_h \otimes W_{h^{-1}g}$$

and so the monoidal structure on the vector space part matches up. Next recall that the tensor product on half braidings is given on a simple object $U_g \in \mathsf{Vec}^{\omega^{-1}} G$ by

$$(\overline{\alpha} \otimes \overline{\beta})_g = \omega^{-1}_{g,V,W} \circ (\overline{\alpha}_g \otimes 1) \circ (\omega^{-1}_{V,g,W})^{-1} \circ (1 \otimes \overline{\beta}_g) \circ \omega^{-1}_{V,W,g}.$$

Similarly, the half braiding from the tensor product of $D^\omega G$ modules is

$$\overline{\mu}_g = F \circ (\mu_g^{-1} \otimes 1)$$

where

$$\begin{aligned}
\mu_g &= (\rho \otimes \psi) \circ \Delta \left(\sum_{h \in G} \delta_h \overline{g}\right) \\
&= (\rho \otimes \psi) \left(\sum_{h,k \in G} \gamma_g(k, k^{-1}h) \delta_k \overline{g} \otimes \delta_{k^{-1}h} \overline{g}\right) \\
&= \sum_{h,k \in G} \gamma_g(k, k^{-1}h) \rho(\delta_k \overline{g}) \otimes \psi(\delta_{k^{-1}h} \overline{g}) \\
&= \sum_{l,m \in G} \gamma_g(l, m) \rho(\delta_l \overline{g}) \otimes \psi(\delta_m \overline{g}).
\end{aligned}$$

In order to prove that these are equal we will prove that their inverses are equal. Hence, consider these acting on a sample element $(u \otimes v \otimes w) \in U_g \otimes V \otimes W$. Additionally, assume that v and w split as $v = \sum_g v_g$ and $w = \sum_g w_g$ with $v_h \in V_h$ and $w_k \in W_k$. Then

$$
\begin{aligned}
((\overline{\alpha}\otimes\overline{\beta})_g)^{-1}(u \otimes v \otimes w) &= (\omega^{-1}_{V,W,g})^{-1} \circ (1 \otimes \overline{\beta}_g) \circ \omega^{-1}_{V,g,W} \circ (\overline{\alpha}_g \otimes 1) \circ (\omega^{-1}_{g,V,W})^{-1}(u \otimes v \otimes w) \\
&= \sum_{h,k} (\omega^{-1}_{V,W,g})^{-1} \circ (1 \otimes \overline{\beta}_g^{-1}) \circ \omega^{-1}_{V,g,W} \circ (\overline{\alpha}_g^{-1} \otimes 1) \circ (\omega^{-1}_{g,V,W})^{-1}(u \otimes v_h \otimes w_k) \\
&= \sum_{h,k} \omega_{g,h,k} (\omega^{-1}_{V,W,g})^{-1} \circ (1 \otimes \overline{\beta}_g^{-1}) \circ \omega^{-1}_{V,g,W} \circ ((\alpha_g \otimes 1) \circ F \otimes 1) \circ (u \otimes v_h \otimes w_k) \\
&= \sum_{h,k} \frac{\omega_{g,h,k}}{\omega_{V,g,k}} (\omega^{-1}_{V,W,g})^{-1} \circ (1 \otimes \beta_g \otimes 1) \circ F \circ (\alpha_g(v_h) \otimes u \otimes w_k) \\
&= \sum_{h,k} \frac{\omega_{g,h,k}\omega_{ghg^{-1},gkg^{-1},g}}{\omega_{ghg^{-1},g,k}} \alpha_g(v_h) \otimes \beta_g(w_k) \otimes u \\
&= \sum_{h,k,l,m} \gamma_g(ghg^{-1}, gkg^{-1}) \rho(\delta_l \overline{g}, v_h) \otimes \psi(\delta_m \overline{g}, w_k) \otimes u \\
&= \sum_{h,k,l,m} \gamma_g(ghg^{-1}, gkg^{-1}) \rho(\delta_l \overline{g}, v_h) \otimes \psi(\delta_m \overline{g}, w_k) \otimes u \\
&= \sum_{l,m} \gamma_g(l, m) \rho(\delta_l \overline{g}, v_{g^{-1}lg}) \otimes \psi(\delta_m \overline{g}, w_{g^{-1}kh}) \otimes u \\
&= \mu_g(v \otimes w) \otimes u \\
&= \overline{\mu}_g(u \otimes v \otimes w).
\end{aligned}
$$

Therefore the equivalence preserves the operation of tensor product on objects. It is easy to show that the operation of tensor product on morphisms is also preserved. It remains to show that Eq preserves the rest of the monoidal structure.

Note that the identity object of $\mathrm{Rep}(D^\omega G)$ is given by $(\mathbb{K}, \epsilon)$ and it can be easily checked that $F(\mathbb{K}, \epsilon) = (V_e, 1)$ where 1 represents the trivial braiding, which is the identity object of $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}} G)^{bop}$. In addition, observe that both the unitors of both categories are trivial. Then using Lemma 2.41, we see that the associator on $\mathrm{Rep}(D^\omega G)$, Φ^{-1} exactly corresponds to multiplying an element in $U_g \otimes V_h \otimes W_k$ by $\omega^{-1}(g, h, k)$. This is exactly the associator on $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}} G)^{bop}$ and so the three Commutative Diagrams 2.3, 2.4 and 2.5 are all satisfied with a trivial tensorator and unit isomorphism. Therefore the previously constructed categorical equivalence is also monoidal equivalence. □

Now we show that this equivalence extends to the braiding.

Proof of Lemma 4.5. Let (V, ρ) and (W, ψ) be two left $D^\omega G$ modules with corresponding objects in $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}} G)^{bop}$, $(V, \overline{\alpha})$ and $(W, \overline{\beta})$.

Recall that there are two possible braidings that can be given to $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}} G)^{bop}$. Both are directly induced from the half braiding and are

$$\sigma_{V,W} = \overline{\alpha}_W$$

and

$$\sigma'_{V,W} = \overline{\beta}_V^{-1} = \sigma^{-1}_{W,V}.$$

On $\mathrm{Rep}(D^\omega G)$, Lemma 2.43 gives the braiding to be $F \circ (\rho \otimes \psi)(R)$. If we try the first braiding, we find that

$$\sigma_{V,W} = \overline{\alpha}_W = F \circ (\alpha_W^{-1} \otimes 1)$$

$$
\begin{aligned}
&= F \circ \sum_{g \in G} \rho\Big(\sum_{h \in G} \delta_h \overline{g}\Big)^{-1} \otimes 1 \\
&= F \circ \sum_{g,h \in G} \theta^{-1}_{ghg^{-1}}(g, g^{-1}) \rho(\delta_h \overline{g^{-1}}) \otimes \psi(\delta_h \overline{e}) \\
&= \sum_{g,h \in G} \theta^{-1}_{ghg^{-1}}(g, g^{-1}) \psi(\delta_h \overline{e}) \otimes \rho(\delta_h \overline{g^{-1}}) \circ F \\
&= (\psi \otimes \rho)(R^{-1}) \circ F.
\end{aligned}
$$

An identical calculation shows that $\sigma'_{V,W} = \sigma^{-1}_{W,V} = F \circ (\rho \otimes \psi)(R)$. Therefore, $\mathrm{Rep}(D^\omega G)$ is braided equivalent to $\mathcal{Z}\big(\mathsf{Vec}^{\omega^{-1}} G\big)^{bop}$ where the braiding is given by the inverse of the half braidings. □

While usually Drinfeld centres are equipped with the opposite braiding to what we have equipped $\mathcal{Z}\big(\mathsf{Vec}^{\omega^{-1}} G\big)^{bop}$ with, fundamentally this is an arbitrary choice and will only have a minor effect later on.

Finally we need to show that the rigid and pivotal structures are also equivalent under this map.

Proof of Lemma 4.6. There are two parts to this theorem. Initially, we must show that the duals with their evaluation and co-evaluation maps are equivalent. Then we can move on the the pivotal structure. Recall that both these structures on $\mathcal{Z}\big(\mathsf{Vec}^{\omega^{-1}} G\big)^{bop}$ are identical to the structures on $\mathsf{Vec}^{\omega^{-1}} G$ discussed in the last chapter.

Dual objects must be preserved by the equivalence as being duals is a property of a pair of object. We just need to show then that our choices for evaluation and co-evalutaion maps also align. Recall that the two constants α and β from the definition of a quasi Hopf algebra are given as $\alpha = \mathbf{1}$ and $\beta = \sum_g \omega(g, g^{-1}, g) \delta_g \overline{e}$. The means that the evaluation map is exactly function application and the co-evaluation map is given by $(1 \otimes \rho^*(\beta)) \circ \eta_V$ where, picking any basis v_i for V, η_V is the map

$$1 \overset{\eta_V}{\longmapsto} \sum_i v_i \otimes v^i.$$

Observe that $S(\beta) = \beta^{-1} = \sum_g \omega^{-1}(g, g^{-1}, g) \delta_g \overline{e}$. Hence, this $\rho^*(\beta)$ term will exactly multiply the g-th graded piece by $\omega^{-1}(g, g^{-1}, g)$ and so this exactly corresponds to the rigid structure for right duals as defined last chapter. Therefore the equivalence preserves the right rigid structure. We will deal with the left rigid structure after showing that the equivalence preserves the pivotal isomorphism.

To find the pivotal structure on $\mathrm{Rep}(D^\omega G)$ we first make the observation that for all $h \in D^\omega G$,

$$S^2(h) = \beta^{-1} h \beta.$$

This means that $\beta^{-1} h = S^2(h) \beta^{-1}$. Then, letting $\overline{\phi}$ be the usual pivotal structure on Vec and choosing an arbitrary $f \in V^*$ we find that

$$
\begin{aligned}
\overline{\phi} \circ \rho(\beta^{-1}) \circ \rho(h)(v)(f) &= \rho(\beta^{-1} h, v)^{**}(f) \\
&= f\Big(\rho(S^2(h)\beta^{-1}, v)\Big) \\
&= \rho^*(S(h), f)\big(\rho(\beta^{-1}, v)\big) \\
&= \big(\rho(\beta^{-1}, v)\big)^* * \Big(\rho^*(S(h), f)\Big) \\
&= \rho^{**}\Big(h, \big(\rho(\beta^{-1}, v)\big)^{**}\Big)(f)
\end{aligned}
$$

$$= \rho^{**}(h) \circ \overline{\phi} \circ \rho(\beta^{-1})(v)(f).$$

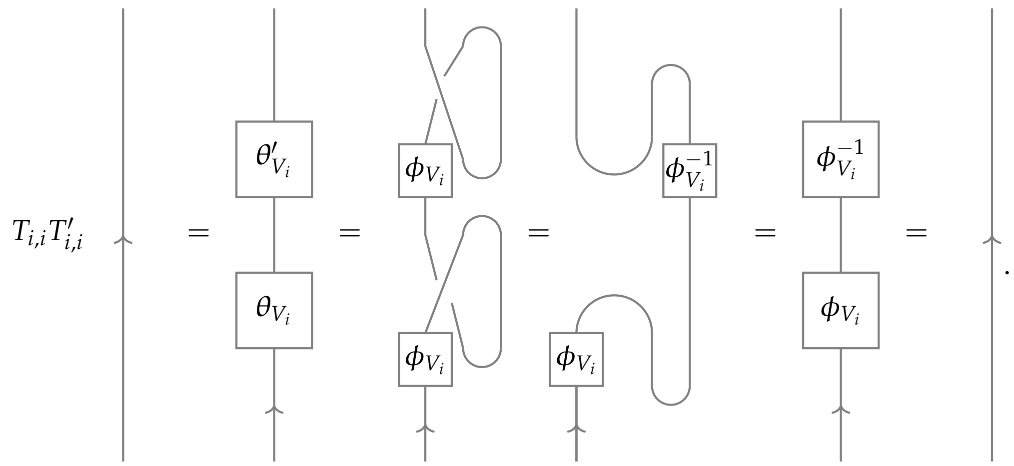
Hence, the pivotal structure on $\mathrm{Rep}(D^\omega G)$ is given by $\overline{\phi} \circ \rho(\beta^{-1})$. As $\beta^{-1} = \sum_g \omega(g, g^{-1}, g)^{-1}\delta_g \bar{e}$ this is exactly multiplying the component of v in V_g by $\omega^{-1}(g, g^{-1}, g)$ and so recalling the pivotal structure on $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$ defined in the last chapter, clearly the equivalence extends to to these pivotal structures.

Recall that the pivotal isomorphism gives a canonical isomorphism between the left and right duals. Hence, as the equivalence preserves the right rigid structure and and the pivotal isomorphism it must also preserve the left rigid structure. □

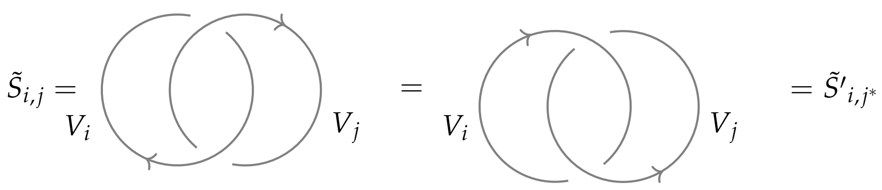
Over the course of these four lemma's we have shown that the equivalence between $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$ and $\mathrm{Rep}(D^\omega G)$ is a braided pivotal equivalence. Further an equivalence of categories will also preserve properties. Therefore as $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$ is ribbon and fusion so is $\mathrm{Rep}(D^\omega G)$. This proves Theorem 4.2 and in particular implies the following Corollary which was in essence the point of defining this Hopf algebra in the first place.

Corollary 4.8. *The S and T matrices of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$ and $\mathrm{Rep}(D^\omega G)$ are equivalent up to a re-ordering of the simple objects.*

Note that it doesn't make sense to ask for the S and T to be equal as it is always possible to relabel the simple objects which certainly will have no effect on the overall category. Let S' and T' denote that S and T matrices of $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)$. It seems reasonable that there should be some connection between S and T and S' and T'. Indeed we can show that $S' = S^{-1}$ and $T' = T^{-1}$. Observe that the string diagram for $T'_{i,i}$ can be placed inside the graphical calculus $\mathcal{Z}\left(\mathsf{Vec}^{\omega^{-1}} G\right)^{bop}$ as it corresponds to the value of the twist θ using the inverse crossing. Then we have, through string manipulations,



Therefore $T'_{i,i} = T_{i,i}^{-1}$ and, as T is diagonal this shows that $T' = T^{-1}$. We can do a similar trick for the S matrix where we rotate the right loop around by 180 degrees out of the page.



In this diagram, V_{j^*} refers to the simple object dual to V_j. It turns out that $S^2 = \frac{\tilde{S}^2}{|G|^2}$ is equal to a matrix C, called the charge conjugation matrix. For the cases we are dealing with, C

is exactly the permutation matrix corresponding to the operation of swapping every simple object with its dual. Hence, recalling that $S^4 = I$, the above diagram manipulations show that $S^{-1} = S^3 = SC = S'$. Therefore the modular data that we calculate for $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$ will be the inverse of the modular data for $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)$.

4.3 Deriving the S and T matrices for $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}})^{bop}$

The equivalence proved above allows us to smoothly move between $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}})^{bop}$ and $\mathrm{Rep}(D^\omega G)$ which will allow us to translate categorical calculations into algebraic ones. This will allow us to derive the formulas for the entries of the S and T matrices shown in Theorem 4.1.

I recommend this moment as the perfect time to make a cup of tea or coffee (or maybe even a little whisky). The following sections will get a little intense at times.

4.3.1 Classifying the Simple Objects

With the equivalence between $\mathrm{Rep}(D^\omega G)$ and $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$, we can use what we know about representations of $D^\omega G$, to analyse $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$. The end goal is of course to produce the S and T matrices of $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$ and so the first step is to classify the simple objects of $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$.

For each conjugacy class $K \subset I(G)$, fix an element $a \in K$.

Theorem 4.9. *There is a bijection between the simple objects of $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$ and the set of tuples (K, a, ρ_a) where K is conjugacy class of G with representative a and ρ_a is an irreducible θ_a-projective representation of the centralizer $C(a)$.*

Proof. This theorem is easier to prove using the equivalence just shown and answering this question in $\mathrm{Rep}(D^\omega G)$. Recall that the simple objects of $\mathrm{Rep}(D^\omega G)$ are exactly the irreducible left modules. Then, as observed earlier, the algebra $D^\omega G$ splits over the conjugacy classes $I(G)$ with

$$D^\omega G = \bigoplus_{K \in I(G)} D^\omega(K, G).$$

Therefore an irreducible left module of $D^\omega G$ must entirely sit over exactly one of these $D^\omega(K, G)$. Consider the subalgebra $D^\omega(a, C(a))$ sitting inside $D^\omega(K, G)$. Observe that this subalgebra is exactly the θ_a-twisted group algebra of $C(a)$ also known as $\mathbb{C}_{\theta_a}[C(a)]$ discussed at the end of Section 2.2.4. Therefore irreducible left modules of $D^\omega(a, C(a))$ exactly correspond to irreducible θ_a-projective representations of $C(a)$.

Next, observe that if (V, ρ) is a left $D^\omega(K, G)$ module, then $V_a = \mathrm{Im}(\rho(\delta_a \bar{e}))$ is a left $D^\omega(a, C(a))$ module with action given by $\rho_a = \rho|_{D^\omega(a,C(a)) \times V_a}$.

<u>Claim:</u> This left module (V_a, ρ_a) is irreducible if and only if (V, ρ) is irreducible.

Proof. Suppose (V_a, ρ_a) is not irreducible. Then there must exist some decomposition $V_a = W_a \oplus W'_a$ such that W_a and W'_a are non-trivial, disjoint and fixed by the action of $D^\omega(a, C(a))$. Now consider the subspace W of V formed by the action of $D^\omega(K, G)$ on W_a. If W is equal to V this would imply the existence of an element $h \in D^\omega(K, G)$ and a $w \in W_a$ such that $h \cdot w \in W'_a$. But such an h would then be contained in $D^\omega(a, C(a))$ as it must send $V_a \to V_a$ and so W_a would not be fixed by the action of $D^\omega(a, C(a))$. This is a contradiction and so W cannot be equal to V. Therefore as W is clearly non-empty, and is fixed by the action of $D^\omega(G)$, it must be a proper submodule of V and therefore (V, ρ) is not irreducible.

Conversely, suppose that (V, ρ) is not irreducible. Then there exists a W a proper subspace of V which is fixed by the action of $D^\omega(K, G)$. Then recall that the dimension of W_k is fixed for all $k \in K$. Hence as W is a proper subspace of V, each W_a must be a proper subspase of V_a. Additionally, as W is fixed by the action of $D^\omega(K, G)$, W_a must be fixed by the action of $D^\omega(a, C(a))$. Therefore W_a is a proper submodule of V_a and so V_a is not irreducible. □

This proves that this restriction map gives a map from the irreducible left modules of $D^\omega(K, G)$ to the irreducible left modules of $D^\omega(a, C(a))$ which, as stated earlier is exactly the irreducible θ_a-projective representations of $C(a)$.

Importantly, this restriction to V_a has an inverse. Given a left $D^\omega(a, C(a))$ module (V_a, ρ_a), the corresponding $D^\omega(K, G)$ module is constructed as follows.[3] Pick any $k \in K$ and consider the subvector space W_k of $D^\omega(K, G)$ generated by elements $\delta_k \overline{g_k}$ with $a = g_k^{-1} k g_k$. This vector space W_k has a right action of $D^\omega(a, C(a))$ given by ∇ and a left action of $D^\omega(k, C(k))$. Hence we define

$$V_k = W_k \otimes_{D^\omega(a,C(a))} V_a.$$

Then, $D^\omega(K, G)$ has a left action on

$$V = \bigoplus_{k\in K} V_k = \bigoplus_{k\in K} W_k \otimes_{D^\omega(a,C(a))} V_a.$$

induced by the action of $D^\omega(K, G)$ on $\oplus_k W_k$.

Recalling that all left modules (V, ρ) of $D^\omega(K, G)$ all admit a decomposition from the images of idempotents $\delta_g \bar{e}$ as

$$V = \bigoplus_{k\in K} \mathrm{Im}\left(\rho(\delta_k \bar{e})\right) = \bigoplus_{k\in K} V_k.$$

It should be clear that this construction is an inverse to the restriction map from $V \mapsto V_a$ defined earlier. Therefore every irreducible left module of $D^\omega(K, G)$ corresponds to an irreducible θ_a-projective representation (V_a, ρ_a) of $C(a)$. □

This has proven the first statement in Theorem 4.1.

4.3.2 Deriving the S Matrix

In order to derive the S matrix, we first need to find the corresponding morphism inside $\mathrm{Rep}(D^\omega G)(I \to I)$. Let (K, a, ρ_a) and (L, b, ψ_b) be two simple objects in $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}} G)^{bop}$. These simple objects will be unambiguously referred to by the vector spaces V and W which correspond to the left modules of $D^\omega G$ with action given by ρ and ψ. Then, the value of $\tilde{S}_{(K,a,\rho_a),(L,b,\psi_b)}$ is the morphism in $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}} G)^{bop}(I \to I)$ given by Figure 4.1. Note that unitors in Figure 4.1 have been omitted because they are trivial.

In order to compute the value of the morphism in Figure 4.1 we will shift everything across to $D^\omega G$ and use algebraic manipulations. In order to achieve this we must first move all of the morphisms over, these have all been covered in the previous chapter but I will restate the equivalences here. For any three left modules (U, τ), (V, ρ) and (W, ψ) we have

$$\eta_V \cong (1 \otimes \rho(\beta)) \circ \eta \qquad 1 \mapsto \sum_i \sum_{g\in G} \omega(g, g^{-1}, g) \rho(\delta_g \bar{e}, v_i) \otimes v^i \tag{4.13}$$

[3]This construction is equivalent to an induction functor. But the work required to construct the category that we are inducting up to is exactly defining all of these W_k. Hence in this case it is easier to directly induct the representation as opposed to using the induction functor.

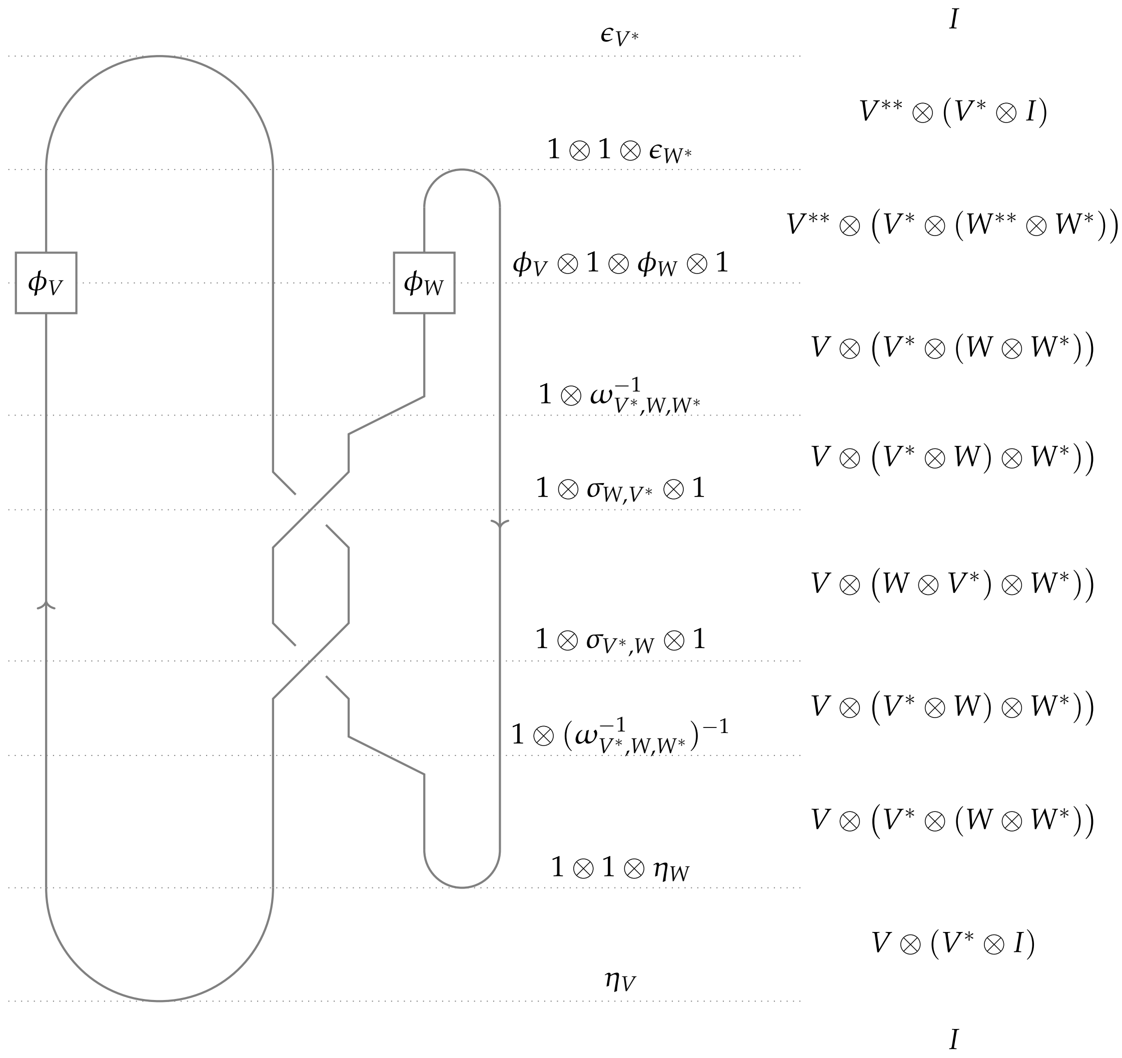


Figure 4.1: A detailed string diagram for deriving the entry of the S matrix corresponding to the simple objects (K, a, ρ_a) and (L, b, ψ_b).

$$\omega_{U,V,W}^{-1} \cong (\tau \otimes \rho \otimes \psi)\Phi^{-1} \qquad u \otimes v \otimes w \mapsto \sum_{g,h,k} \frac{1}{\omega(g,h,k)} \tau(\delta_g \overline{e}, u) \otimes \tau(\delta_h \overline{e}, v) \otimes \tau(\delta_k \overline{e}, w) \tag{4.14}$$

$$\sigma_{V,W} \cong \tau \circ (\rho \otimes \psi)(R) \qquad v \otimes w \mapsto \sum_{g,h \in G} \psi(\delta_h \overline{g}, w) \otimes \rho(\delta_g \overline{e}, v) \tag{4.15}$$

$$\phi_V \cong \phi \circ \rho(\beta^{-1}) \qquad v \mapsto \sum_{g} \frac{1}{\omega(g, g^{-1}, g)} \big(\rho(\delta_g \overline{e}, v)\big)^{**} \tag{4.16}$$

$$\epsilon_V \cong \epsilon \qquad f \otimes v \mapsto f(v). \tag{4.17}$$

Let us start by calculating the action of the double braiding. This will be easier now than if we tried to do it directly in the larger morphism. This double braiding corresponds to the map on (V, ρ), (W, ψ) given by

$$\sigma_{V,W}^2 = \tau \circ (\psi \otimes \rho)(R) \circ \tau \circ (\rho \otimes \psi)(R).$$

This map acts on a pair $v \otimes w \in V \otimes W$ as follows

$$\begin{aligned} \tau \circ (\psi \otimes \rho)(R) \circ \tau \circ (\rho \otimes \psi)(R)(v \otimes w) &= \tau \circ (\psi \otimes \rho)(R) \Big(\sum_{g,h \in G} \psi(\delta_h \overline{g}, w) \otimes \rho(\delta_g \overline{e}, v) \Big) \\ &= \sum_{g,h,k,l \in G} \rho\big(\delta_l \overline{k}, \rho(\delta_g \overline{e}, v)\big) \otimes \psi\big(\delta_k \overline{e}, \psi(\delta_h \overline{g}, w)\big). \end{aligned}$$

As ψ and ρ are representations, composition is equivalent to multiplication. Thus, as $\theta_g(x, e) = \theta_g(e, x) = 1$ for all $g, x \in G$,

$$\rho(\delta_l \overline{k}) \rho(\delta_g \overline{e}) = \begin{cases} \rho(\delta_k g k^{-1} \overline{k}) & \text{if } l = kgk^{-1} \\ 0 & \text{else} \end{cases}$$

and

$$\psi(\delta_k \overline{e}) \psi(\delta_h \overline{g}) = \begin{cases} \psi(\delta_h \overline{g}) & \text{if } k = h \\ 0 & \text{else.} \end{cases}$$

Thus we can remove the sums over k and l, replacing l with hgh^{-1} and k with h. Additionally, replace g with hgh^{-1}. This gives the double braiding to be

$$\sigma_{V,W}^2 = \sum_{g,h \in G} \Big(\psi(\delta_g \overline{h})(v), \rho(\delta_h \overline{h^{-1} g h})(w) \Big).$$

The simplifications performed in the above calculation will be continually used to simplify the more complex expressions that will appear when calculating the coefficients of the S matrix.

Let us now jump right in to computing the S matrix morphism. Note that $\mapsto$ will denote the steps where a new morphism is being applied and $=$ will correspond to simplification steps.

$$\begin{aligned} 1 &\overset{\eta_V}{\longmapsto} \sum_{\substack{i \\ g \in G}} \frac{\omega(g, g^{-1}, g)}{1} \rho(\delta_g \overline{e}, v_i) \otimes v^i \\ &\xmapsto{1 \otimes 1 \otimes \eta_W} \sum_{\substack{i,j \\ g,h \in G}} \frac{\omega(g, g^{-1}, g) \omega(h, h^{-1}, h)}{1} \rho(\delta_g \overline{e}, v_i) \otimes v^i \otimes \rho(\delta_h \overline{e}, w_j) \otimes w^j \end{aligned}$$

$$\xmapsto{1\otimes(\omega^{-1}_{V^*,W,W^*})^{-1}} \sum_{\substack{i,j\\ g,hk,l,m\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,l,m)}{1}\rho(\delta_g\overline{e},v_i)\otimes\rho^*(\delta_k\overline{e},v^i)$$
$$\otimes\,\psi\big(\delta_l\overline{e},\psi(\delta_h\overline{e},w_j)\big)\otimes\psi^*\big(\delta_m\overline{e},w^j\big)$$
$$=\sum_{\substack{i,j\\ g,h,k,m\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,m)}{1}\rho(\delta_g\overline{e},v_i)\otimes\rho^*(\delta_k\overline{e},v^i)$$
$$\otimes\,\psi(\delta_h\overline{e},w_j)\otimes\psi^*\big(\delta_m\overline{e},w^j\big)$$
$$\xmapsto{1\otimes\sigma^2_{V,W}\otimes 1} \sum_{\substack{i,j\\ g,h,k,m,p,q\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,m)}{1}\rho(\delta_g\overline{e},v_i)\otimes\rho^*\big(\delta_q\overline{p}\rho^*(\delta_k\overline{e},v^i)\big)$$
$$\otimes\,\psi\Big(\delta_p\overline{p^{-1}qp},\psi(\delta_h\overline{e},w_j)\Big)\otimes\psi^*\big(\delta_m\overline{e},w^j\big)$$
$$=\sum_{\substack{i,j\\ g,h,k,m\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,m)}{1}\rho(\delta_g\overline{e},v_i)\otimes\rho^*(\delta_{khkh^{-1}k^{-1}}\overline{khk^{-1}},v^i)$$
$$\otimes\,\psi(\delta_{khk^{-1}}\overline{k},w_j)\otimes\psi^*\big(\delta_m\overline{e},w^j\big)$$
$$\xmapsto{1\otimes\omega^{-1}_{V^*,W,W^*}} \sum_{\substack{i,j\\ g,h,k,m,l,p,q\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(l,p,q)}\rho(\delta_g\overline{e},v_i)$$
$$\otimes\,\rho^*\big(\delta_l\overline{e},\rho^*(\delta_{khkh^{-1}k^{-1}}\overline{khk^{-1}},v^i)\big)\otimes\psi\big(\delta_p\overline{e},\psi(\delta_{khk^{-1}}\overline{k},w_j)\big)\otimes\psi^*\big(\delta_q\overline{e},\psi^*\big(\delta_m\overline{e},w^j\big)\big)$$
$$=\sum_{\substack{i,j\\ g,h,k,m\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(khkh^{-1}k^{-1},khk^{-1},m)}\rho(\delta_g\overline{e},v_i)\otimes\rho^*(\delta_{khkh^{-1}k^{-1}}\overline{khk^{-1}},v^i)$$
$$\otimes\,\psi(\delta_{khk^{-1}}\overline{k},w_j)\otimes\psi^*\big(\delta_m\overline{e},w^j\big)$$
$$\xmapsto{\phi_V\otimes 1\otimes\phi_W\otimes 1} \sum_{\substack{i,j\\ g,h,k,m,p,q\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(khkh^{-1}k^{-1},khk^{-1},m)\omega(p,p^{-1},p)\omega(q,q^{-1},q)}\Big(\rho\big(\delta_p\overline{e},\rho(\delta_g\overline{e},v_i)\big)\Big)^{**}$$
$$\otimes\,\rho^*(\delta_{khkh^{-1}k^{-1}}\overline{khk^{-1}},v^i)\otimes\Big(\psi\big(\delta_q\overline{e},\psi(\delta_{khk^{-1}}\overline{k},w_j)\big)\Big)^{**}\otimes\psi^*\big(\delta_m\overline{e},w^j\big)$$
$$=\sum_{\substack{i,j\\ g,h,k,m\in G}} \frac{\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(khkh^{-1}k^{-1},khk^{-1},m)\omega(khk^{-1},kh^{-1}k^{-1},khk^{-1})}\Big(\rho(\delta_g\overline{e},v_i)\Big)^{**}$$
$$\otimes\,\rho^*(\delta_{khkh^{-1}k^{-1}}\overline{khk^{-1}},v^i)\otimes\Big(\psi(\delta_{khk^{-1}}\overline{k},w_j)\Big)^{**}\otimes\psi^*\big(\delta_m\overline{e},w^j\big)$$
$$\xmapsto{\epsilon_{V^*}\circ(1\otimes 1\otimes\epsilon_{W^*})} \sum_{\substack{i,j\\ g,h,k,m\in G}} \frac{\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(khkh^{-1}k^{-1},khk^{-1},m)\omega(khk^{-1},kh^{-1}k^{-1},khk^{-1})}$$
$$\times\Big(\rho(\delta_g\overline{e},v_i)\Big)^{**}\Big(\rho^*(\delta_{khkh^{-1}k^{-1}}\overline{khk^{-1}},v^i)\Big)\times\Big(\psi(\delta_{khk^{-1}}\overline{k},w_j)\Big)^{**}\Big(\psi^*\big(\delta_m\overline{e},w^j\big)\Big)$$
$$=\sum_{\substack{i,j\\ g,h,k,m\in G}} \frac{\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(khkh^{-1}k^{-1},khk^{-1},m)\omega(khk^{-1},kh^{-1}k^{-1},khk^{-1})}$$
$$\times\,\rho^*(\delta_{khkh^{-1}k^{-1}}\overline{khk^{-1}},v^i)\big(\rho(\delta_g\overline{e},v_i)\big)\times\psi^*(\delta_m\overline{e},w^j)\big(\psi(\delta_{khk^{-1}}\overline{k},w_j)\big)$$

To shrink this expression slightly let, let $q=khkh^{-1}k^{-1}$ and $p=khk^{-1}$. Then substituting this in, and applying the definitions of the dual actions gives

$$\sum_{\substack{i,j \\ g,h,k,m\in G}} \frac{\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(q,p,m)\omega(p,p^{-1},p)} \times v^i\Big(\rho\big(S(\delta_q\overline{p}),\rho(\delta_g\overline{e},v_i)\big)\Big)w^j\Big(\psi\big(S(\delta_m\overline{e}),\psi(\delta_p\overline{k},w_j)\big)\Big)$$

Next, apply the definition of the antipode S to produce the expression

$$\sum_{\substack{i,j \\ g,h,k,m\in G}} \frac{\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(q,p,m)\omega(p,p^{-1},p)\theta_{q^{-1}}(p,p^{-1})\gamma_p(q,q^{-1})}\times$$
$$v^i\Big(\rho\big(\delta_{p^{-1}q^{-1}p}\overline{p^{-1}},\rho(\delta_g\overline{e},v_i)\big)\Big)w^j\Big(\psi\big(\delta_{m^{-1}}\overline{e},\psi(\delta_p\overline{k},w_j)\big)\Big)$$

Then, $\rho\big(\delta_{p^{-1}q^{-1}p}\overline{p^{-1}},\rho(\delta_g\overline{e},v_i)\big)$ will be zero unless $p^{-1}q^{-1}p = p^{-1}gp$ which is clearly equivalent to the equality $g = q^{-1}$. Similarly, in order for $\psi\big(\delta_{m^{-1}}\overline{e},\psi(\delta_p\overline{k},w_j\big)$ to be non-zero we must have that $m = p^{-1}$. Substituting these equalities in brings us to the equation

$$\sum_{\substack{i,j \\ g,h,k,m\in G}} \frac{\omega(h,h^{-1},h)\omega(k,h,m)}{\omega(g^{-1},m^{-1},m)\omega(m^{-1},m,m^{-1})\theta_g(m^{-1},m)\gamma_{m^{-1}}(g^{-1},g)}\times$$
$$v^i\big(\rho(\delta_{mgm^{-1}}\overline{m},v_i)\big)w^j\big(\psi(\delta_{m^{-1}}\overline{k},w_j)\big).$$

Choose a basis for V and W that diagonalises the action of $\delta_g\overline{e}$ for all $g\in G$. Then $w^j\big(\psi(\delta_{m^{-1}}\overline{k},w_j)\big)$ will be zero unless $k\in C(m)$ and w_j lies in $W_{m^{-1}}$. As $m = p^{-1} = kh^{-1}k^{-1}$, this means that $m = h^{-1}$. Therefore k and h must commute and so $g = q^{-1} = khk^{-1}h^{-1}k^{-1} = k^{-1}$. Additionally, in order for $v^i\big(\rho(\delta_{mgm^{-1}}\overline{m},v_i\big) = v^i\big(\rho(\delta_g\overline{h^{-1}},v_i\big)$ to be non-zero, it is required that $v_i\in V_g$ and $h\in C(g)$. Let w_{h_j} be a basis for W_h and let v_{g_i} be a basis of V_g. Adding in all of these conditions simplifies the expression further to

$$\sum_{\substack{g\in G \\ h\in C(g) \\ g_i,h_j}} \frac{1}{\theta_g(h,h^{-1})\gamma_h(g^{-1},g)} v^{g_i}\big(\rho(\delta_g\overline{h^{-1}},v_{g_i})\big)w^{h_j}\big(\psi(\delta_h\overline{g^{-1}},w_{h_j})\big).$$

As V and W correspond to irreducible $D^\omega G$ modules they are entirely concentrated over conjugacy classes K and L. Then $\sum_{g_i} v^{g_i}\big(\rho(\delta_g\overline{h^{-1}},v_{g_i})\big)$ exactly gives the trace of the matrix $\rho(\delta_g\overline{h^{-1}})$ and, as $\rho(\delta_g\overline{h^{-1}})$ acts by 0 outside V_g,

$$\mathrm{tr}\left(\rho(\delta_g\overline{h^{-1}})\right) = \mathrm{tr}\left(\rho_g(h^{-1})\right) = \chi_{\rho_g}(h^{-1}).$$

Additionally, observe that when g and h commute, $\gamma_h(g^{-1},g) = \theta_h(g^{-1},g)$. Plugging all these in further simplifies the expression to

$$\sum_{\substack{g\in K \\ h\in KL\cap C(g)}} \frac{1}{\theta_g(h,h^{-1})\theta_h(g^{-1},g)}\chi_{\rho_g}(h^{-1})\chi_{\psi_h}(g^{-1}).$$

This interpretation as projective characters allows us to use some projective character theory. In particular we can use Equation 2.30 which gives

$$\chi_{\rho_g}(h^{-1}) = \theta_g(h,h^{-1})\chi^*_{\rho_g}(h).$$

Using this equality and observing that the 2-cocycle condition for normalised cocycles implies that $\theta_h(g^{-1}, g) = \theta_h(g, g^{-1})$, we can simplify the expression for the S matrix component further to

$$\sum_{\substack{g \in K \\ h \in L \cap C(g)}} \chi^*_{\rho_g}(h)\chi^*_{\psi_h}(g).$$

Unfortunately, while this is the simplest version of the formula, it cannot be used for computations. This is because we will only have the representations ρ_a and ψ_b. Therefore we need to find a way to express $\chi^*_{\rho_g}(h)$ in terms of a ρ_a character.

Let y be an element conjugate to a and $h \in C(y)$. Then pick an element a_y in G that witnesses the conjugacy of a and y. That is to say, $a = a_y y a_y^{-1}$. Then consider the following calculation

$$\begin{aligned}
\chi_{\rho_y}(h) &= \text{tr}\left(\rho(\delta_y \overline{h})\right) \\
&= \theta_y(a_y^{-1}, a_y)^{-1}\, \text{tr}\left(\rho\big(\nabla(\nabla(\delta_y \overline{a_y^{-1}}, \delta_a \overline{a_y}), \delta_y \overline{h})\big)\right) \\
&= \theta_y(a_y^{-1}, a_y)^{-1}\, \text{tr}\left(\rho(\delta_y \overline{a_y^{-1}})\rho(\delta_a \overline{a_y})\rho(\delta_y \overline{h})\right) \\
&= \theta_y(a_y^{-1}, a_y)^{-1}\, \text{tr}\left(\rho(\delta_a \overline{a_y})\rho(\delta_y \overline{h})\rho(\delta_y \overline{a_y^{-1}})\right) \\
&= \theta_y(a_y^{-1}, a_y)^{-1}\, \text{tr}\left(\rho\big(\nabla(\nabla(\delta_a \overline{a_y}, \delta_y \overline{h}), \delta_y \overline{a_y^{-1}})\big)\right) \\
&= \frac{\theta_a(a_y, h)\theta_a(a_y h, a_y^{-1})}{\theta_y(a_y^{-1}, a_y)}\, \text{tr}\left(\rho(\delta_a \overline{a_y h a_y^{-1}})\right) \\
&= \frac{\theta_a(a_y, h)\theta_a(a_y h, a_y^{-1})}{\theta_y(a_y^{-1}, a_y)} \chi_{\rho_a}(a_y h a_y^{-1}).
\end{aligned} \tag{4.18}$$

Hence given a fixed $a \in K$ and $b \in L$ for each other $g \in K$ and $h \in L$ pick an a_g and b_h such that $a = a_g h a_g^{-1}$ and $b = b_h h b_h^{-1}$. Then, applying the formula we just derived to current expression of the S matrix gives us

$$S_{(K,a,\rho_a),(L,b,\psi_b)} = \sum_{\substack{g \in K \\ h \in L \cap C(g)}} \left(\frac{\theta_a(a_g, h)\theta_a(a_g h, a_g^{-1})\theta_b(b_h, g)\theta_b(b_h g, b_h^{-1})}{\theta_g(a_g^{-1}, a_g)\theta_h(b_h^{-1}, b_h)} \right)^* \chi^*_{\rho_a}(a_g h a_g^{-1})\chi^*_{\psi_b}(b_h g b_h^{-1}).$$

Then, simply dividing through by the normalisation constant, ($|G|$) will give the formula for the S matrix as stated in Theorem 4.1.

4.3.3 Deriving the T Matrix

In order to find the T matrix let us draw another detailed picture as we did for the S matrix. Let (K, a, ρ_a) be a simple object with corresponding left module (V, ρ), then $|V|T_{(K,a,\rho_a)}$ is given by the morphism in Figure 4.2.

We find the corresponding morphism in the same manner as before.

$$\begin{aligned}
1 &\xmapsto{\eta_{V^*}} \sum_{\substack{i \\ g \in G}} \omega(g, g^{-1}, g)\rho^*(\delta_g \overline{e}, v_i^*) \otimes v_i^{**} \\
&\xmapsto{1 \otimes 1 \otimes \eta_V} \sum_{\substack{i,j \\ g,h \in G}} \omega(g, g^{-1}, g)\omega(h, h^{-1}, h)\rho^*(\delta_g \overline{e}, v_i^*) \otimes v_i^{**} \otimes \rho(\delta_h \overline{e}, v_j) \otimes v_j^*
\end{aligned}$$

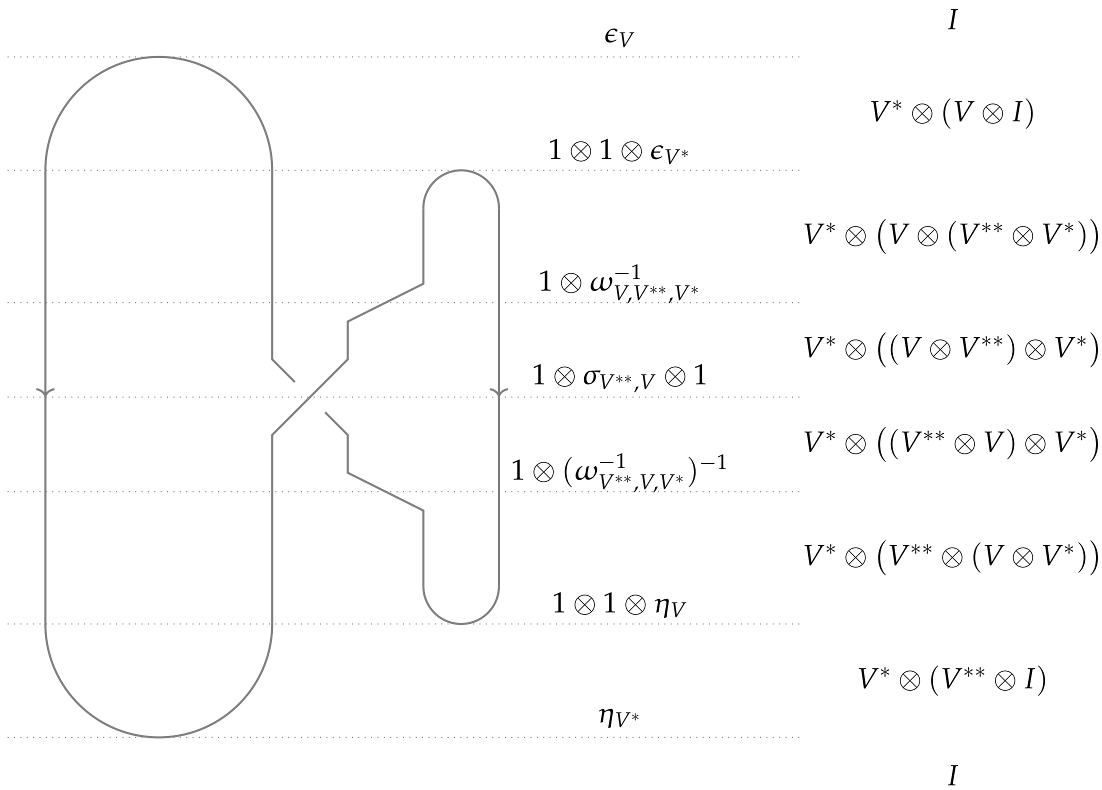


Figure 4.2: A detailed string diagram for deriving the diagonal entry of the T matrix corresponding to the simple object (K, a, ρ_a).

$$\xmapsto{1\otimes(\omega^{-1}_{V^{**},V,V^*})^{-1}} \sum_{\substack{i,j\\ g,h,k,l\in G}} \omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,l)\rho^*(\delta_g\bar{e},v_i^*)$$
$$\otimes \rho^{**}(\delta_k\bar{e},v_i^{**})\otimes\rho(\delta_h\bar{e},v_j)\otimes\rho^*(\delta_l\bar{e},v_j^*)$$
$$\xmapsto{1\otimes\sigma_{V^{**},V}} \sum_{\substack{i,j\\ g,h,k,l,m,n\in G}} \omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,l)\rho^*(\delta_g\bar{e},v_i^*)\otimes\rho\big(\big(\delta_n\overline{m},\rho(\delta_h\bar{e},v_j)\big)$$
$$\otimes\rho^{**}\big(\delta_m\bar{e},\rho^{**}(\delta_k\bar{e},v_i^{**})\big)\otimes\rho^*(\delta_l\bar{e},v_j^*)$$
$$=\sum_{\substack{i,j\\ g,h,k,l\in G}} \omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,l)\rho^*(\delta_g\bar{e},v_i^*)\otimes\rho(\delta_{khk^{-1}}\bar{k},v_j)$$
$$\otimes\rho^{**}(\delta_k\bar{e},v_i^{**})\otimes\rho^*(\delta_l\bar{e},v_j^*)$$
$$\xmapsto{1\otimes\omega^{-1}_{V,V^{**}}} \sum_{\substack{i,j\\ g,h,k,l\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,l)}{\omega(khk^{-1},k,l)}\rho^*(\delta_g\bar{e},v_i^*)\otimes\rho(\delta_{khk^{-1}}\bar{k},v_j)$$
$$\otimes\rho^{**}(\delta_k\bar{e},v_i^{**})\otimes\rho^*(\delta_l\bar{e},v_j^*)$$
$$\xmapsto{\epsilon_V\circ(1\otimes 1\otimes\epsilon_{V^*})} \sum_{\substack{i,j\\ g,h,k,l\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,l)}{\omega(khk^{-1},k,l)}\rho^*(\delta_g\bar{e},v_i^*)\big(\rho(\delta_{khk^{-1}}\bar{k},v_j)\big)$$
$$\times\rho^{**}(\delta_k\bar{e},v_i^{**})\big(\rho^*(\delta_l\bar{e},v_j^*)\big)$$
$$=\sum_{\substack{i,j\\ g,h,k,l\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,l)}{\omega(khk^{-1},k,l)} v_i^*\Big(\rho\big(S(\delta_g\bar{e}),\rho(\delta_{khk^{-1}}\bar{k},v_j)\big)\Big)$$
$$\times v_i^{**}\Big(\rho^*\big(S(\delta_k\bar{e}),(\rho^*(\delta_l\bar{e},v_j^*)\big)\Big)$$
$$=\sum_{\substack{i,j\\ g,h,k,l\in G}} \frac{\omega(g,g^{-1},g)\omega(h,h^{-1},h)\omega(k,h,l)}{\omega(khk^{-1},k,l)} v_i^*\Big(\rho\big(\delta_{g^{-1}}\bar{e},\rho(\delta_{khk^{-1}}\bar{k},v_j)\big)\Big)$$
$$\times v_i^{**}\Big(\rho^*\big(\delta_{k^{-1}}\bar{e},(\rho^*(\delta_l\bar{e},v_j^*)\big)\Big)$$
$$=\sum_{\substack{i,j\\ h,k\in G}} \frac{\omega(kh^{-1}k^{-1},khk^{-1},kh^{-1}k^{-1})\omega(h,h^{-1},h)\omega(k,h,k^{-1})}{\omega(khk^{-1},k,k^{-1})} v_i^*\Big(\rho(\delta_{khk^{-1}}\bar{k},v_j)\Big)v_i^{**}\Big(\rho^*(\delta_{k^{-1}}\bar{e},v_j^*)\Big)$$

Observe now that

$$v_i^{**}\Big(\rho^*(\delta_{k^{-1}}\bar{e},v_j^*)\Big)=v_j^*\big(\rho(S(\delta_{k^{-1}}\bar{e}),v_i)\big)=v_j^*\big(\rho(\delta_k\bar{e},v_i)\big).$$

Hence this term is non-zero only if $j=i$ and $v_i\in V_k$ in which case, this term is 1. Let v_{k_i} be a basis of V_k, then subbing this all in, the expression simplifies to

$$\sum_{\substack{h,k\in G\\ k_i}} \frac{\omega(kh^{-1}k^{-1},khk^{-1},kh^{-1}k^{-1})\omega(h,h^{-1},h)\omega(k,h,k^{-1})}{\omega(khk^{-1},k,k^{-1})} v_{k_i}^*\Big(\rho(\delta_{khk^{-1}}\bar{k},v_{k_i})\Big)$$

Then, this term can only be non-zero if $khk^{-1}=k$ which clearly implies that $h=k$. Then, the expression simplifies further to

$$\sum_{\substack{k\in G\\ k_i}} \frac{\omega(k^{-1},k,k^{-1})\omega(k,k^{-1},k)\omega(k,k,k^{-1})}{\omega(k,k,k^{-1})} v_{k_i}^*\Big(\rho(\delta_k \overline{k}, v_{k_i})\Big).$$

Observe that as ω is normalized, the 3-cocycle condition on (k,k^{-1},k,k^{-1}) implies that

$$\omega(k^{-1},k,k^{-1})\omega(k,k^{-1},k) = 1.$$

Next, recall the chosen element $a \in K$ and pick an $a_k \in G$ such that $a = a_k k a_k^{-1}$. Then treating $\sum_{k_i}\Big(\rho(\delta_k\overline{k}, v_{k_i})\Big)$ as a character and using Equation 4.18 again to replace χ_{ρ_k} with a χ_{ρ_a} we get

$$T_{(K,a,\rho_a),(L,b,\psi_b)} = \frac{\delta_{K,L}\delta_{\rho_a,\psi_b}}{|V|}\sum_{k\in K}\chi_{\rho_k}(k) = \frac{\delta_{K,L}\delta_{\rho_a,\psi_b}}{|V|}\chi_{\rho_a}(a)\sum_{k\in K}\frac{\theta_a(a_k,k)\theta_a(a_k k, a_k^{-1})}{\theta_k(a_k^{-1},a_k)}.$$

Consider further the term

$$\frac{\theta_a(a_k,k)\theta_a(a_k k, a_k^{-1})}{\theta_k(a_k^{-1},a_k)}.$$

First observe that Equation 4.10 with $g = k$, $x = a_k^{-1}$, $y = a_k^{-1}$, and $z = k$ gives the equality

$$\theta_k(a_k^{-1},a_k)\theta_k(e,k) = \theta_k(a_k^{-1},a_k k)\theta_{a_k k a_k^{-1}}(a_k,k).$$

As θ is normalised, and $a_k k a_k^{-1} = a$, this summation term can be rearranged by

$$\begin{aligned}
\frac{\theta_a(a_k,k)\theta_a(a_k k, a_k^{-1})}{\theta_k(a_k^{-1},a_k)} &= \frac{\theta_a(a_k k, a_k^{-1})}{\theta_k(a_k^{-1},a_k k)}\\
&= \frac{\omega(a,a_k,k,a_k^{-1})\omega(a_k k,a_k^{-1},a_k k^{-1}a_k^{-1}a a_k k a_k^{-1})\omega(a_k^{-1},a_k k a_k^{-1},a_k k)}{\omega(a_k k,k^{-1}a_k^{-1}a a_k k,a_k^{-1})\omega(k,a_k^{-1},a_k k)\omega(a_k^{-1},a_k k,k^{-1}a_k^{-1}a_k k a_k^{-1}a_k k)}\\
&= \frac{\omega(a_k k a_k^{-1},a_k,k,a_k^{-1})\omega(a_k k,a_k^{-1},a_k k a_k^{-1})\omega(a_k^{-1},a_k k a_k^{-1},a_k k)}{\omega(a_k k,k,a_k^{-1})\omega(k,a_k^{-1},a_k k)\omega(a_k^{-1},a_k k,k)}
\end{aligned}$$

This expression appears complicated but it can be simplified using the 3-cocycle condition. Applying the 3-cocycle condition on the tuple $(a_k k, a_k^{-1}, a_k k, a_k^{-1})$ simplifies this expression to

$$\frac{\omega(a_k k,a_k^{-1},a_k,k)\omega(a_k^{-1},a_k k,a_k^{-1})\omega(a_k^{-1},a_k k a_k^{-1},a_k k)}{\omega(k,a_k^{-1},a_k k)\omega(a_k^{-1},a_k k,k)}$$

which the 3-cocycle condition on the tuple $(a_k^{-1}, a_k k, a_k^{-1}, a_k k)$ shows must be 1. Therefore, recalling that $|V| = |K||V_a| = |K|\chi_{\rho_a}(e)$ we are left with the simple formula

$$T_{(K,a,\rho_a),(L,b,\psi_b)} = \delta_{K,L}\delta_{\rho_a,\psi_b}\frac{\chi_{\rho_a}(a)}{\chi_{\rho_a}(e)}. \tag{4.19}$$

This completes the proof of Theorem 4.1.

Chapter 5

Computing Modular Data in GAP

As described previously, given a group G and a unitary 3-cocycle ω, there is a corresponding modular tensor category $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$. This section details code created to construct the S and T matrices of $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$ from a given G and ω. This will then be used to create a database of S and T matrices for all $|G| < 48$ with a unique ω taken from every orbit of $H^3(G, \mathbb{C})$ under the action of the automorphism group $\operatorname{Aut}(G)$. Symmetry arguments show that if $\{\omega_i\}$ is a complete set of representatives of $H^3(G, \mathbb{C})/\operatorname{Aut}(G)$ then $\{\omega_i^{-1}\}$ is also a complete set of representatives. Thus as the modular data for $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$ is the inverse of the modular data for $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)$, the constructed database will contain inverted modular data for every equivalence class of categories of the form $Z(\mathsf{Vec}^{\omega}G)$ with $|G|$ less than 48.

The programming language GAP [20],was chosen for this project due to both its speed and inbuilt functions. In particular, GAP can easily create groups from a given presentation and find the corresponding character tables. On top of GAP there were two packages used, 'hap' and 'IO'. The package 'hap' [21], adds in numerous functions relating to group cohomology and was required for the code written by Michaël Mignard and Peter Schauenburg. The package 'IO' [22] was used to aid the back end of storing the computed modular data. It added in background file manipulations such as the ability to open and close files on demand.

The method of creating the database of modular data can be broken down into three distinct steps.

The first step is to produce a compete set of representatives of $H^3(G, \mathbb{C})/\operatorname{Aut}(G)$. Initially we must compute $H^3(G, \mathbb{C})$. This is done using the Universal Coefficient Theorem and was implemented by Michaël Mignard and Peter Schauenburg in their recent paper [1]. Their code produces a basis for the cohomology space and then represents each unique cocycle by a vector. Then, they have written a function to find the orbits of the cohomology space under the action of $\operatorname{Aut}(G)$. For most groups this worked fine but there a couple of groups for which their code was too slow. For groups of order 2^n, my supervisor Scott Morrison wrote some bit-flipping code that computed the orbits more quickly. Currently we have not expanded this to groups whose orders are not powers of 2 but this would be an important exercise if we wanted to increase the size of the modular data database.

The code then moves on to constructing the list of simple objects of $\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)^{bop}$. This is done by finding the set of conjugacy classes of G. Then, for each conjugacy class K, choosing a representative a and computing the 2-cocycle θ_a on $C(a)$. Using the group extension from Theorem 2.52, we construct the θ_a-character table of $C(a)$ and each row of this character table corresponds to a unique simple object.

Finally, the code needs to implement the pair of formulas found in Theorem 4.1 to produce the S and T matrices. In practice, computing the S matrix using Equation 4.3 is slow. However, it is possible to make several speed improvements using the numerous symmetries of the S matrix. See Lemma 5.1 and the discussion following it.

Some of the code given here has been modified for brevity. The full functions can be found in the Appendix for an interested reader. They are also available at `https://tqft.net/web/research/students/AngusGruen` along with the computed modular data to download and use.

5.1 Choosing a Set of 3-cocycles

The following lines of code implement the algorithm written by Michaël Mignard and Peter Schauenburg on a given group G. The variable `orbits` will then be a list of vectors corresponding to a single cocycles from each orbit of $H^3(G,\mathbb{C})/\operatorname{Aut}(G)$.

```
R           := ResolutionFiniteGroup(G,4);
K           := TensorWithIntegers(R);
UCT         := UniversalCoefficientsTheorem(K, 3);
orbits      := AutomorphismOrbits(G, R, K, UCT);
orbits      := List(orbits, orbit -> UCT.lift*orbit[1]);
```

Given such a vector, the following lines of code turn it into the 3-cocycle that will be used in the rest of the code.

```
omega := StandardCocycle(R, vector, 3, UCT.exponent);
val   := Gcd(Gcd(vector), UCT.exponent);
exp   := UCT.exponent/val;
```

Here `omega` is a map from $G \times G \times G \to \{0, \cdots, \texttt{UCT.exponent} - 1|\} \subset \mathbb{Z}$ where `UCT.exponent` is the LCM of the torsion coefficients of $H^3(G,\mathbb{C})$. Next, `val` is the GCD of elements in the image of `omega`, identifying 0 with `UCT.exponent` and `exp` is `UCT.exponent/val`. The reason for computing `val` and `exp` is that all elements in the image of `omega` will be divisible by val. Therefore when viewed as a map into $\mathbb{C}^\times$,

$$\texttt{omega/val} : G \times G \times G \to \{0, \cdots, \texttt{exp} - 1\}$$

is the same map as `omega` but, due to its smaller image, this second map will be quicker to compute with.

With this data, the following function will produce the S and T matrices.

```
ComputeModularData := function(G, omega, val, exponent)
  local simples;

  simples  := GenerateSimpleObjects(G, omega, val, exponent);

  return GenerateModularData(G, omega, val, exponent, simples);
end;
```

5.2 Computing the Simple Objects

The function `GenerateSimpleObjects` begins by producing the set of conjugacy classes of G. Then, for each conjugacy class K, it will pick a representative a and create a list of pairs[1] $[g, a_g]$

[1] Recall that a_g is chosen so that $a = a_g g a_g^{-1}$.

for every element in K. Next it constructs the the group extension of $C(a)$ using the function `GroupExtensionOfCentralizer` and then finds all the θ_a-irreducible characters using the function `ProjectiveCharacters`. Then, a list of simple objects is created with every pair of a conjugacy class and an irreducible projective representation corresponding to a unique simple object. For each simple object we record 4 pieces of data. We record the representative element of the conjugacy class, the projective character of the projective representation, the list of elements $[g, a_g]$ described earlier and the dimension of the projective representation. For speed purposes, the projective character is stored as a lookup dictionary linking each element of the centralizer to the projective character of that element.

```
GenerateSimpleObjects := function(G, omega, val, exponent)
  local simples, conjClass, i, g, j, dict, h, reps, extensionData,
        iso, Cg, projCharData, EmbedIntoPCGroupExtCg;

  simples     := [];
  conjClass   := ConjugacyClasses(G);

  for i in conjClass do
    g := Representative(i);
    reps := [];
    for j in i do
      Add(reps, [j, RepresentativeAction(G, g, j)]);
    od;

    extensionData         := GroupExtensionOfCentralizer
                                (G, g, omega, val, exponent);
    iso                   := extensionData[1];
    Cg                    := extensionData[2];
    projCharData          := ProjectiveCharacters(
                                 extensionData[3].CgExtGens,
                                 extensionData[3].CgExtPc,
                                 extensionData[3].CgExtIso);
    EmbedIntoPCGroupExtCg := extensionData[3].embedding;

    for j in projCharData do

      dict := NewDictionary(false, true, Cg);

      for h in Cg do
        AddDictionary(dict, h,
        EmbedIntoPCGroupExtCg(h^iso)^j);
      od;

      Add(simples, rec( representative := g,
        character := dict,
        conjugacyClass := reps,
        dimension := j[1]));
    od;
  od;
  return simples;
```

```
end;
```

Note that this is a slightly simplified version of my code. In actual fact, the case where the conjugacy class is $\{e\}$ is treated differently for two reasons. Firstly as $\theta_e = 1$, the projective representations correspond to ordinary representations of G which can be computed quickly. Additionally, this guarantees that the first simple object is the unit object and so the first row and column of the S matrix will be strictly positive rational numbers and correspond to a list of the normalised dimensions of each simple object. The normalised dimension of a simple object V in a category $\mathscr{C}$ is $\frac{\dim(V)}{\sqrt{\dim(\mathscr{C})}}$.

There are two functions that I wrote called here, `GroupExtensionOfCentralizer` and `ProjectiveCharacters`. `GroupExtensionOfCentralizer` creates the centralizer of g as a finitely presented group and specializes the 3-cocycle ω to a 2-cocycle θ_g which is called `bbeta` in the code. Inside `GroupExtensionOfCentralizer`, the function `GroupExtension` is called to create $C(g)_{\theta_g}$ as discussed in Section 2.2.4. This function is long and not particularly illustrative and so has been placed in Appendix B as opposed to being given here.

The function `GroupExtension` creates $C(g)_{\theta_g}$ by finding a finite presentations for it. The finite presentation used is presented in more detail in [23]. Let $\langle g_i | r_j \rangle$ be a finite presentation of G. Then define $\overline{\theta_g}$ to be a map from words over the alphabet $\{g_i, g_i^{-1}\}$ to $\{0, \cdots, \exp - 1\}$ by

$$\overline{\theta_g}(g_{i_1}^{e_1} \cdots g_{i_n}^{e_n}) = \prod_{j=1}^{n} \theta_g(g_{i_1}^{e_1} \cdots g_{i_{j-1}}^{e_{j-1}}, g_{i_j}^{e_j}) \theta_g(g_{i_j}, g_{i_j}^{-1})^{\frac{e_i - 1}{2}}.$$

A finite presentation for G_{θ_g} is given by

$$G_{\theta_g} = \left\langle g_i, x | x^{\exp}, x g_i x^{-1} g_i^{-1}, r_j x^{-\overline{\theta_g}(r_j)} \right\rangle.$$

Comparing this to the description of this group in Section 2.2.4 in terms of $G \otimes C_{exp}$ with a twisted multiplication, observe that the generator g_i here corresponds to the element $(g_i, 1) \in G \otimes C_{exp}$ and the generator x corresponds to (e, x).

`GroupExtension` returns a record containing the generators of $C(g)_{\theta_g}$, an isomorphism from $C(g)_{\theta_g}$ to a polycyclic group if $C(g)_{\theta_g}$ is solvable or a permutation group if it is not, the image of this isomorphism and an embedding function from $C(g)$ through $C(g)_{\theta_g}$ into its isomorphic polycyclic/permutation group. The reason for representing $C(g)_{\theta_g}$ as a polycyclic/permutation group is because it will speed up the calculation of the character table of the group.

```
GroupExtensionOfCentralizer := function(G, g, omega, val, exponent)
  local centralizer, iso, beta, bbeta;
  centralizer := Centralizer(G, g);
  iso         := IsomorphismFpGroup(centralizer);
  beta        := TwoCocycleOnCentralizer(omega, g);

  bbeta       := function(h, k)
    return beta(PreImage(iso, h), PreImage(iso, k))/val;
  end;

  return [iso, centralizer,
          GroupExtension(Image(iso), exponent, bbeta)];
end;
```

`GroupExtensionOfCentralizer` will return a list with 3 elements. The first is an isomorphism from a finitely presented version of $C(g)$ to $C(g)$. The second is $C(g)$ and the third is the record returned by `GroupExtension`.

After constructing the group extension we then use it to find the θ_g-projective representations using the function ProjectiveCharacters as described in Section 2.2.4.

```
ProjectiveCharacters := function(CgExtGens, CgExtPc, iso)
  local irr, elm, exp, ident;
  irr   := Irr(CgExtPc);
  elm   := CgExtGens[Length(CgExtGens)]^iso;
  exp   := Order(elm);
  ident := Identity(CgExtPc);

  return Filtered(irr, i -> elm^i = (ident^i)*E(exp));
end;
```

This function is fairly self-explanatory. It first constructs a list of irreducible characters of $C(g)_{\theta_g}$ and finds the element (e, x) inside $C(g)_{\theta_g}$ which is given by the last generator of the finite presentation. Then it checks the condition

$$\frac{\chi(e,x)}{\chi(e,1)} = e^{\frac{2\pi i}{Order(x)}}$$

on every irreducible character. The characters that satisfy this correspond to θ_g-projective irreducible characters and so are returned.

It is important to note that some of the inbuilt GAP functions used here are non-deterministic. This means that running this piece of code twice will likely produce different permutations of the list of simple objects. In particular this means that the S and T matrices produced will not necessarily by identical. They will of course be equivalent up to conjugative by a permutation matrix.

After generating the simple objects, the code moves on to creating the S and T matrices using the function `GenerateModularData`.

5.3 Computing the S and T Matrices

The code below is a simplification of what my code does and there are several large optimisations that I have removed for clarity. These optimisations are discussed after Lemma 5.1 later in this section. The full function can also be found in Appendix B.[2] Essentially we directly implement the formulas from Theorem 4.1. The function `CoefficientSSum` is exactly calculating the coefficient

$$\frac{\theta_a(a_g, h)\theta_a(a_g h, a_g^{-1})\theta_b(b_h, g)\theta_b(b_h g, b_h^{-1})}{\theta_g(a_g^{-1}, a_g)\theta_h(b_h^{-1}, b_h)}.$$

Then, `sVal` is exactly implements Equation 4.3 when given 2 simple objects. To make the S matrix, we iterate over the created list of simple objects twice and call `sVal` on each pair to create the corresponding entry of the matrix. The T matrix is comparatively easier to make as we only need to compute the diagonal. Therefore we only need to loop over the simple objects once, directly implementing Equation 4.2.

[2] As a warning, the function is several pages long.

```
CoefficientSSum := function(exponent, val, beta, a, b, h, l, xh, yl)
  local int;

  int := beta(a, xh, l) + beta(a, xh*l, xh^-1)
         + beta(b, yl, h) + beta(b, yl*h, yl^-1)
         - beta(h, xh^-1, xh) - beta(l, yl^-1, yl);

  return E(exponent)^(int/val);
end;


sVal := function(G, exponent, val, beta, gData, hData)
  local num, ggg, hhh, iii, jjj, coeff, charVal;

  num := 0;
  ggg := gData.representative;
  hhh := hData.representative;

  for iii in gData.conjugacyClass do
   for jjj in hData.conjugacyClass do
    if iii[1]*jjj[1] = jjj[1]*iii[1] then
      coeff   := CoefficientSSum(exponent, val, beta,
          ggg, hhh, iii[1], jjj[1], iii[2], jjj[2]);
      charVal := LookupDictionary(gData.character,
              iii[2]*jjj[1]*(iii[2]^-1))*
              LookupDictionary(hData.character,
              jjj[2]*iii[1]*(jjj[2]^-1));
      num     := num + coeff*charVal;
    fi;
  od; od;

  return ComplexConjugate(num) / Order(G);
end;


GenerateModularData := function(G, omega, val, exponent, simples)
  local tMatrixDiagonal, sMatrix, beta;

  beta := function(g, h, k)
    return omega(g, h, k) + omega(h, k, (k^-1)*(h^-1)*g*h*k)
           - omega(h, (h^-1)*g*h, k);
  end;

  tMatrixDiagonal := List(simples, simple ->
      LookupDictionary(simple.character, simple.representative)/
          simple.dimension);

  sMatrix := List(simples, simple1 ->
                    List(simples, simple2 ->
```

```
                                    sVal(G, exponent, val, beta, simple1, simple2)))

  return rec(S := sMatrix, T := DiagonalMat(tMatrixDiagonal));
end;
```

In practise, this code can be sped up a lot. This comes from the fact that calling `sVal` is expensive and the S matrix has several symmetries.

The first trick is observing that when G is abelian, all conjugacy classes contain only 1 element and all centralizers are the entire group. This simplifies the formula for elements of the S matrix to be

$$S_{(\{g\},g,\rho_g),(\{h\},h,\psi_h)} = \chi^*_{\rho_g}(h)\chi^*_{\psi_h}(g).$$

Additionally, as $S = S^T$ we only need to compute the upper triangular half of S. Then, the S matrix for abelian groups can be quickly implemented by two nested for loops.

When G is not abelian, more elaborate methods are needed. We use the following result found in [10],

Lemma 5.1. *Given any row S_x of the S matrix, and any element $\sigma \in Gal(\mathbb{Q}[S_x]/\mathbb{Q})$, denote $\sigma(S_x)$, the list given by applying σ to every element of S_x. Then either $\sigma(S_x)$ or $-\sigma(S_x)$ also appears as a row in the S matrix.*

For our specific case, as S_1 is always a positive rational, $-\sigma(S_x)$ will never be a row and so $\sigma(S_x)$ will always be one. Additionally, $\mathrm{Gal}(\mathbb{Q}[S_x]/\mathbb{Q}) \subset \mathrm{Gal}(\mathbb{Q}[T]/\mathbb{Q})$ and so this group can be precomputed after calculating the diagonal of the T-matrix.

This allows for the following algorithm. Start with an empty S-matrix. Then repeat the following steps:

1. Compute a random[3] unknown row and add it to the S matrix.

2. If this computed row is not a Galois conjugate of an already computed row, save all of its Galois conjugates to a list containing other rows that are Galois conjugates of computed rows.

3. Go through the list of saved Galois conjugates and attempt to place them into the partially filled S matrix using the constraint that $S = S^T$. If there is only a single place that a row could go, place it there.

4. If the S matrix is not yet complete, return to step 1.

In practise this algorithm can be hundreds of times faster than the direct approach. Exactly what the speed differential is depends on the specific group and 3-cocycle but even for small groups is it usually at least twice as fast.

Finally, there is some background code that saves the computed data to a GAP parseable file as well as running some tests on it. All of this background code can be found at `https://tqft.net/web/research/students/AngusGruen`.

[3]The reason for computing a random row each time as opposed to going through the list of rows deterministically is because we want to avoid computing rows that are equal to Galois conjugates of rows already computed. In practise it is usually the case that many of the Galois conjugates of a row are the preceding and ensuing rows. Therefore it is a lot faster in general to compute a Random row each time than go through the list of rows deterministically.

Chapter 6

Analysis of Modular Data

While the main aim of this project was to construct the database and thus allow other people to do more detailed analysis, there is a small amount of analysis that we do here. One interesting question relates to the ranks of $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$ given the order of G. Recall that the rank of $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$ is exactly the dimension of the S and T matrices. Interestingly, for the data that we have constructed, the ranks appear to be heavily restricted by the order of G. As $\dim\left(\mathcal{Z}(\mathsf{Vec}^{\omega}G)\right) = |G|^2$ we know that $\operatorname{rank}\left(\mathcal{Z}(\mathsf{Vec}^{\omega}G)\right)$ is bounded by $|G|^2$ but, in general, only few of the values in the range $\{1, \cdots, |G|^2\}$ are obtained. Figure 6.1 gives the possible ranks of $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$ for a fixed value of $|G|$ it also gives the multiplicities of these ranks with respect to the equivalence classes of $\mathsf{Vec}^{\omega}G$ and the equivalences of the modular data of $\mathcal{Z}(\mathsf{Vec}^{\omega}G)$.

There are some clear patterns that can be seen in these results. For example, for all odd primes p, there is only 1 group of order p which has exactly 3 orbits of $H^3(G,\mathbb{C})/\operatorname{Aut}(G)$ each of which corresponds to a different Morita equivalence class. Note that we know that each orbits corresponds to a different Morita equivalence class because the numbers in columns 3 and 4 are identical. Additionally, when p is two times an odd prime, there are 2 groups of order p both with 6 orbits of $H^3(G,\mathbb{C})/\operatorname{Aut}(G)$ with each orbit corresponding to a to a different Morita equivalence class. There also appears to be a a pattern when $|G|$ is three times a prime larger than 3.

As for some more general observations, at every order $|G|$ numerous categories have rank $|G|^2$. This is to be expected as for any abelian group G, $\operatorname{rank}\left(\mathcal{Z}(\mathsf{Vec}\,G)\right) = |G|^2$. Interestingly though, for the majority of groups calculated, the rank of the centre is independent of the cocycle chosen. This may be an effect of small groups however as, as $|G|$ increases, the percentage of groups exhibiting at least 2 different ranks of their centre appears to rise.

Another interesting feature is that, for all groups so far computed, $\operatorname{rank}\left(\mathcal{Z}(\mathsf{Vec}\,G)\right)$ is an upper bound to $\operatorname{rank}\left(\mathcal{Z}(\mathsf{Vec}^{\omega^{-1}}G)\right)$. It is clear that this should be the case for Abelian groups, but it is far less clear if this should be expected to hold for non-Abelian groups.

We can also use these calculations to get a lower bound on the number of Morita equivalence classes of pointed fusion categories of a fixed dimension. Two categories with modular data S, T and S', T' cannot be Morita equivalent unless there exists a permutation matrix P such that $S = PS'P^{-1}$ and $T = PT'P^{-1}$. However it is non trivial to find such a P matrix or show that one does not exist. In order to solve this, we convert the problem to a graph isomorphism question and then use the program `nauty` written by Brendan Mackay and Adolfo Piperno [24]. This program `nauty` accepts a pair of graphs and returns either an automorphism or states that no automorphism exists. How to convert a matrix to an edge coloured graph is explained on page 60 of the user manual: `http://pallini.di.uniroma1.it/nug26.pdf`. To test if S and S' are equivalent, we convert them both to edge coloured graphs making sure that the colours correspond to the same values and them give the pair of graphs to `nauty`.

Table 6.2, details the upper and lower bounds that we have constructed. The naive upper

$\lvert G \rvert$	Ranks of the Center	Multiplicities up to regular equivalence	Multiplicities up to modular data equivalence
2	4	2	2
3	9	3	3
4	16	8	7
5	25	3	3
6	8, 36	6, 6	6, 6
7	49	3	3
8	22, 64	25, 22	20, 18
9	81	10	9
10	16, 100	6, 6	6, 6
11	121	3	3
12	14, 32, 144	6, 30, 24	6, 27, 21
13	169	3	3
14	28, 196	6, 6	6, 6
15	225	9	9
16	46, 88, 256	66, 189, 73	47, 125, 58
17	289	3	3
18	44, 72, 324	20, 18, 20	18, 18, 18
19	361	3	3
20	22, 64, 400	4, 30, 24	4, 27, 21
21	25, 441	3, 9	3, 9
22	64, 484	6, 6	6, 6
23	529	3	3
24	21, 42, 56, 86, 128, 198, 576	24, 36, 12, 141, 120, 75, 66	24, 30, 12, 114, 99, 60, 54
25	625	10	9
26	88, 676	6, 6	6, 6
27	105, 729	31, 30	23, 24
28	112, 784	30, 24	27, 21
29	841	3	3
30	116, 144, 200, 900	18, 18, 18, 18	18, 18, 18, 18
31	961	3	3
32	79, 100, 121, 142, 184, 352, 1024	60, 589, 72, 129, 1978, 1081, 172	60, 330, 72, 94, 1072, 580, 108
33	1089	9	9
34	148, 1156	6, 6	6, 6
35	1225	9	9
36	36, 64, 126, 176, 288, 1296	8, 42, 30, 100, 90, 80	8, 42, 28, 81, 81, 63
37	1369	3	3
38	184, 1444	6, 6	6, 6
39	65, 1521	3, 9	3, 9
40	88, 214, 256, 550, 1600	20, 141, 120, 75, 66	18, 114, 99, 60, 54
41	1681	3	3
42	44, 100, 224, 252, 392, 1764	6, 6, 18, 18, 18, 18	6, 6, 18, 18, 18, 18
43	1849	3	3
44	256, 1936	30, 24	27, 21
45	2025	30	27
46	268, 2116	6, 6	6, 6
47	2209	3	3

Figure 6.1: The distribution of the ranks of $\mathcal{Z}(\mathrm{Vec}^{\omega}G)$ for a fixed order of G between 2 and 47.

bounds originate directly from the number of equivalence classes of $\text{Vec}^{\omega}G$ and the lower bounds are the number of in equivalent pairs of modular data.

For $|G|$ between 2 and 31 this table is identical to a table in Mignard and Schauenburg's paper [1] and in their paper they prove that for $|G| \leq 31$ categories whose centres had equivalent modular data were Morita equivalent. At $|G| = 32$ we have improved on the lower bound given by Mignard and Schauenburg from 2315 to 2316 and for $33 \leq |G| \leq 47$ these lower bounds have never been published before. Due to the exactness of the lower bounds for $|G| \leq 31$, it is likely that the bounds for $|G| \geq 32$ are either optimal or at least very close.

The invariants used by Mignard and Schauenburg were the Frobenius-Schur indicators and the T matrix. Therefore the improvement in the lower bound means that the S and T matrices are strictly stronger invariants than the Frobenius-Schur indicators and the T matrix. Unfortunately, as we do not have the equivalence files produced by Mignard and Schauenburg while we know that an example exists, we cannot yet explicitly give a pair of categories which show this fact. It would be relatively simple to compute the Frobenius-Schur indicators from our data and thus pin down this example exactly but for now this is beyond the scope of this project. While the result that the S and T matrices are strictly stronger invariants than the Frobenius-Schur indicators and the T matrix was already known [25], the counterexample given in [25] involves more exotic categories than the ones presented here and so it would be a useful exercise to explicitly find the counterexample at $|G| = 32$.

There were two effects that prevented us from calculating all of the modular data for groups of order bigger than 47, both caused by abelian groups. The first issue was related to cyclic groups C_n. In generating the simple objects, GAP needs to produce the character tables of $(C(a))_{\theta_a}$ for every $a \in G$. When $G = C_n$, $C(a) = C_n$ and the LCM of the torsion coefficients of $H^3(C_n, \mathbb{C})$ is n. This means that the group extension $(C_n)_{\theta_a}$ will in general[1] be a group of order n^2. For $n = 48$, this group has 2304 elements and computing the character table of this takes a little over half an hour. As there are 48 conjugacy classes of C_n, computing a single set of modular data for these groups can take over a day. The other slowdown originated from groups with large automorphism groups. While computing the cohomology classes remains quick, finding representatives of the $\text{Aut}(G)$ orbits can be very slow. While we could deal with $\left(\mathbb{Z}/2\mathbb{Z}\right)^5$ by writing some bit-flipping code we were unable to extend this to the group $G = \left(\mathbb{Z}/2\mathbb{Z}\right)^4 \times \left(\mathbb{Z}/3\mathbb{Z}\right)$. This meant that we could not find the orbits of $H^3(G, \mathbb{C})/\operatorname{Aut}(G)$. Both of these problems could probably be fixed to some degree with either more work or more computing power and then the database could be extended further possibly even past $|G| = 100$.

[1]The extension can be smaller if `val` is bigger than 1 but for the majority of cases this is not the case.

$\|G\|$	# of Groups	Naive upper bound of # of $\mathcal{Z}(\text{Vec}^{\omega}G)$	Lower Bound of # of $\mathcal{Z}(\text{Vec}^{\omega}G)$
2	1	2	2
3	1	3	3
4	2	8	7
5	1	3	3
6	2	12	12
7	1	3	3
8	5	47	38
9	2	10	9
10	2	12	12
11	1	3	3
12	5	60	54
13	1	3	3
14	2	12	12
15	1	9	9
16	14	328	230
17	1	3	3
18	5	58	54
19	1	3	3
20	5	58	52
21	2	12	12
22	2	12	12
23	1	3	3
24	15	474	393
25	2	10	9
26	2	12	12
27	5	61	47
28	4	54	48
29	1	3	3
30	4	72	72
31	1	3	3
32	51	4081	2316
33	1	9	9
34	2	12	12
35	1	9	9
36	14	350	303
37	1	3	3
38	2	12	12
39	2	12	12
40	14	422	345
41	1	3	3
42	6	84	84
43	1	3	3
44	6	54	48
45	1	30	27
46	6	12	12
47	1	3	3

Figure 6.2: The lower and upper bounds for the number of Morita equivalence classes of pointed fusion categories of ranks 2 through 47.

Group Cohomology

We present the proof of Theorem 2.29, which states that all cocycles are cohomologous to a unitary cocycles.

In order to prove this Theorem it is usually easiest to first prove the following lemma.

Lemma A.1. *Every cocycle is cohomologous to a normalized cocycle.*

Proof. See Section 6.5 of [13]. □

Proof of Theorem 2.29. Due to the Lemma just proved it suffices to prove Theorem 2.29 for normalized cocycles. Thus let α be a normalized n-cocycle. Then define

$$\beta(g_1, \cdots, g_{n-1}) = \prod_{g_n \in G} \alpha(g_1, \cdots, g_{n-1}, g_n).$$

As α is an n-cocycle

$$\begin{aligned} 1 = &\alpha(g_2, \cdots, g_{n+1}) \times \alpha(g_1, \cdots, g_n)^{(-1)^{n+1}} \\ &\times \prod_{i=1}^{n} \alpha(g_1, \cdots, g_{i-1}, g_i g_{i+1}, g_{i+2}, \cdots, g_{n+1})^{(-1)^i}. \end{aligned}$$

Rearranging, we find that

$$\begin{aligned} \left(\alpha(g_1, \cdots, g_n)^{(-1)^n}\right)^{|G|} &= \prod_{g_{n+1} \in G} \Big(\alpha(g_2, \cdots, g_{n+1}) \\ &\qquad \times \prod_{i=1}^{n} \alpha(g_1, \cdots, g_{i-1}, g_i g_{i+1}, g_{i+2}, \cdots, g_{n+1})^{(-1)^i}\Big) \\ &= \beta(g_2, \cdots, g_n) \times \prod_{i=1}^{n-1} \beta(g_1, \cdots, g_{i-1}, g_i g_{i+1}, g_{i+2}, \cdots, g_n)^{(-1)^i} \\ &\qquad \times \beta(g_1, \cdots, g_{n-1})^{(-1)^n} \\ &= \partial^n \beta(g_1, \cdots, g_n). \end{aligned}$$

Hence let β' be a normalized $(n-1)$-cochain such $\beta'^{|G|} = \beta$. Define $\alpha' = \alpha(\partial^n \beta')^{-1}$, clearly α' is cohomologous to α. As β' is normalized so is $d\beta'$ and so as α is also normalized so is α'. Additionally, $\alpha'^{|G|} = \alpha^{|G|}(\partial^n \beta'^{-|G|}) = \partial^n \beta \partial^n \beta^{-1} = 1$ and so α' is unitary. Therefore α is cohomologous to a unitary cocycle. □

More Code

This section contains the code that was omitted from Chapter 5. First we give the complete version of the function `GenerateSimpleObjects`.

```
GenerateSimpleObjects := function(G, omega, val, exponent)
  local simples, conjClass, i, g, j, dict, h, reps,
        extensionData, iso, Cg, projCharData, EmbedIntoPCGroupExtCg;

  simples     := [];
  conjClass   := ConjugacyClasses(G);

  for i in conjClass do
    g := Representative(i);
    if g = Identity(G) then
      for j in Irr(G) do
        dict := NewDictionary(false, true, G);

        for h in G do
            AddDictionary(dict, h, h^j);
        od;

        Add(simples, rec( representative := g,
                          character := dict,
                          conjugacyClass := [[g, g]],
                          dimension := j[1])          );
      od;
    else
      reps := [];
      for j in i do
        Add(reps, [j, RepresentativeAction(G, g, j)]);
      od;

      extensionData           := GroupExtensionOfCentralizer
                                    (G, g, omega, val, exponent);
      iso                     := extensionData[1];
      Cg                      := extensionData[2];
      projCharData            := ProjectiveCharacters(
                                    extensionData[3].CgExtGens,
                                    extensionData[3].CgExtPc,
```

```
                                           extensionData[3].CgExtIso  );

       EmbedIntoPCGroupExtCg   := extensionData[3].embedding;

       for j in projCharData do

         dict := NewDictionary(false, true, Cg);

         for h in Cg do
             AddDictionary(dict,
                           h,
                           EmbedIntoPCGroupExtCg(h^iso)^j);
         od;

         Add(simples, rec( representative := g,
                           character := dict,
                           conjugacyClass := reps,
                           dimension := j[1])     );
       od;
     fi;
   od;
   return simples;
 end;
```

Next we give the functions `GroupExtension` and `ListGenToElement`.

```
GroupExtension := function(Cg, exponent, beta)
    local gens, rels, m, n, RelationTwist, Embedding,
          relations, i, rel, F, gensF, iso, EmbedIntoGroupExtension;

  gens  := GeneratorsOfGroup(Cg);
  rels  := RelatorsOfFpGroup(Cg);
  m     := Length(gens);
  n     := m + 1;


  RelationTwist := function(word)
    local start, i, val;
    start   := Identity(Cg);
    val     := 0;
    for i in word do
      if i > 0 then
        val   := val + beta(start, gens[i]);
        start := start*gens[i];
      else
        val   := val + beta(start, gens[-i]^-1);
        start := start*(gens[-i]^-1);
        val   := val - beta(gens[-i]^-1, gens[-i]);
      fi;
    od;
```

```
    return (-val) mod exponent;
  end;

  Embedding := function(h)
    local word;
    word := LetterRepAssocWord(h);
    return Concatenation(word,
      ListWithIdenticalEntries(RelationTwist(word), n));
  end;

  relations := [ListWithIdenticalEntries(exponent, n)];

  for i in [1..m] do
    Add(relations, [i, n, -i, -n]);
  od;

  for rel in rels do
    Add(relations, Embedding(rel));
  od;

  F     := FreeGroup(n);
  gensF := GeneratorsOfGroup(F);

  relations := List(relations, x -> ListGenToElement(x, F, gensF));
  F         := F / relations;
  gensF     := GeneratorsOfGroup(F);

  if IsSolvableGroup(Cg) then
    iso := IsomorphismPcGroup(F);
  else
    iso := IsomorphismPermGroup(F);
  fi;

  EmbedIntoGroupExtension := function(word)
    local lst, ginChExt;
    lst       := Factorization(Cg, word);
    ginChExt  := ListGenToElement(Embedding(lst), F, gensF);
    return ginChExt^iso;
  end;


  return rec( CgExtGens := gensF,
              CgExtPc := Image(iso),
              CgExtIso := iso,
              embedding := EmbedIntoGroupExtension);
end;

ListGenToElement := function(word, G, gens)
  local v, i;
```

```
  v := Identity(G);

  for i in word do
    if i > 0 then
        v := v*gens[i];
    else
        v := v*((gens[-i])^-1);
    fi;
  od;

  return v;
end;
```

Finally we give the complete function `GenerateModularData` which implements all of the discussed optimisations involving Galois automorphisms.

```
GenerateModularData := function(G, omega, val, exponent, simples)
  local i, j, k, numSimples, tMatrixDiagonal, cyclotomicDegree,
        galoisGroup, beta, sMatrix, rowsToCompute, justComputed,
        mysteryRows, mysteryRowPlacing, nextRow, entry, newRow,
        galoisConjugates, conjugateRow, possible, stillMystery,
        foundPlacing, mystery, possiblePlacings, stillPossible,
        rowsComputed;

  numSimples      := Length(simples);
  tMatrixDiagonal := List(simples, simple ->
      LookupDictionary(simple.character, simple.representative)/
          simple.dimension);

  beta := function(g, h, k)
    return omega(g, h, k)
              + omega(h, k, (k^-1)*(h^-1)*g*h*k)
              - omega(h, (h^-1)*g*h, k);
  end;

  sMatrix := NullMat(numSimples, numSimples);

  if IsAbelian(G) then
    for i in [1..numSimples] do for j in [i..numSimples] do
      entry := ComplexConjugate(LookupDictionary(
                  simples[j].character, simples[i].representative)*
                  LookupDictionary(simples[i].character,
                  simples[j].representative)/Order(G));

      sMatrix[i][j] := entry;
      sMatrix[j][i] := entry;
    od; od;
  else
    cyclotomicDegree    := Conductor(tMatrixDiagonal);
    galoisGroup         := PrimeResidues(cyclotomicDegree);
```

```
rowsToCompute       := [1..numSimples];
rowsComputed        := [];
justComputed        := [];
mysteryRows         := [];
mysteryRowPlacing   := [];

while not rowsToCompute = [] do

  if justComputed = [] then

    nextRow := Remove(rowsToCompute,
        Random(1, Length(rowsToCompute)));
    Add(rowsComputed, nextRow);

    sMatrix[nextRow][nextRow] := sVal(G, exponent, val,
            beta, simples[nextRow], simples[nextRow]);

    for i in rowsToCompute do
      entry := sVal(G, exponent, val, beta,
              simples[nextRow], simples[i]);
      sMatrix[i][nextRow] := entry;
      sMatrix[nextRow][i] := entry;
    od;

    justComputed := [nextRow];
    newRow := sMatrix[nextRow];

    if newRow in mysteryRows then
      i := Position(mysteryRows, newRow);
      Remove(mysteryRows, i);
      Remove(mysteryRowPlacing, i);
    else
      galoisConjugates := [newRow];
      for i in galoisGroup do
        conjugateRow := GaloisCyc(newRow, i);
        if not conjugateRow in galoisConjugates then
          Add(galoisConjugates, conjugateRow);
          Add(mysteryRows, conjugateRow);
          possiblePlacings := [];
          for j in rowsToCompute do
            if ForAll(rowsComputed, k ->
              conjugateRow[k] = sMatrix[j][k])
                  then
              Add(possiblePlacings, j);
            fi;
          od;
          Add(mysteryRowPlacing,
              possiblePlacings);
        fi;
```

```
        od;
      fi;
    fi;

    stillMystery := [];
    foundPlacing := [];

    for i in [1..Length(mysteryRows)] do
      mystery           := mysteryRows[i];
      possiblePlacings  := mysteryRowPlacing[i];
      stillPossible     := [];
      for j in possiblePlacings do
        if ForAll(justComputed, k -> (not j = k)
          and (mystery[k] = sMatrix[k][j])) then
          Add(stillPossible, j);
        fi;
      od;

      if Length(stillPossible) = 1 then
        Add(foundPlacing, [i, stillPossible[1]]);
      else
        mysteryRowPlacing[i] := stillPossible;
        Add(stillMystery, i);
      fi;
    od;

    justComputed := [];

    for i in foundPlacing do
      for j in rowsToCompute do
        sMatrix[i[2]][j] := mysteryRows[i[1]][j];
        sMatrix[j][i[2]] := mysteryRows[i[1]][j];
      od;

    Add(justComputed, i[2]);
    Add(rowsComputed, Remove(rowsToCompute,
            Position(rowsToCompute, i[2])));

    od;

    mysteryRows       := mysteryRows{stillMystery};
    mysteryRowPlacing := mysteryRowPlacing{stillMystery};

  od;
fi;

return rec(S := sMatrix, T := DiagonalMat(tMatrixDiagonal));
end;
```

Bibliography

1. Michaël Mignard and Peter Schauenburg. Morita Equivalence of Pointed Fusion Categories of small rank. arXiv:1708.06538.
2. Jacob Lurie. On the Classification of Topological Field Theories. *Current Developments in Mathematics*, 2008:129–280, 2009. arXiv:0905.0465.
3. Christopher L. Douglas, Christopher Schommer-Pries, and Noah Snyder. Dualizable tensor categories. arXiv:1312.7188.
4. Cain Edie-Michell. Equivalences of graded categories. To appear.
5. Antoine Coste, Terry Gannon, and Philippe Ruelle. Finite Group Modular Data. *Nuclear Physics B*, 581(3):679 – 717, 2000. arXiv:hep-th/0001158.
6. AnRan Chen. On Frobenius-Schur indicators for Spherical Fusion Categories. 2017. Honours Thesis ANU.
7. Pavel Etingof, Shlomo Gelaki, Dmitri Nikshych, and Victor Ostrik. *Tensor Categories*. American Mathematical Society, 2015.
8. S. Mac Lane. *Categories for the Working Mathematician*. Springer-Verlag, 1971.
9. Peter J Freyd and David N Yetter. Braided compact closed categories with applications to low dimensional topology. *Advances in Mathematics*, 77(2):156 – 182, 1989. DOI:10.1016/0001-8708(89)90018-2.
10. A. Coste and T. Gannon. Remarks on Galois symmetry in rational conformal field theories. *Physics Letters B*, 323(3):316 – 321, 1994. DOI:10.1016/0370-2693(94)91226-2.
11. Joyal Andre and Ross Street. Tortile Yang-Baxter operators in tensor categories. 71, 04 1991.
12. Michael Müeger. From subfactors to categories and topology II. *Journal of Pure and Applied Algebra*, 180(1):159–219, 2003. arXiv:math/0111205.
13. Charles A Weibel. *An Introduction to Homological Algebra*. Cambridge University Press, 1994.
14. Vyjayanthi Chari and Andrew Pressley. *A guide to Quantum Groups*. Cambridge University Press, 1994.
15. Chuangxun Cheng. A character theory for projective representations of finite groups. *Linear Algebra and its Applications*, 469(Supplement C):230 – 242, 2015. DOI:10.1016/j.laa.2014.11.027.
16. Dmitri Nikshych, Pavel Etingof, and Viktor Ostrik. On fusion categories. *Annals of Mathematics*, 162:581–642, 2005. arXiv:math/0203060.
17. Robert Coquereaux. Character tables (modular data) for Drinfeld doubles of finite groups. arXiv:1212.410.

18. Robbert Dijkgraff and Edward Witten. Topological gauge theories and group cohomology. *Communications in Mathematical Physics*, 129(2):393–429, 1990. URL `https://projecteuclid.org/euclid.cmp/1104180750`.

19. P. Bantay. Orbifolds, Hopf Algebras, and the Moonshine. *Letters in Mathematical Physics*, 22 (3):187–194, Jan 1991. DOI:10.1007/BF00403544.

20. GAP. *GAP – Groups, Algorithms, and Programming, Version 4.8.8*. The GAP Group, 2017. URL `https://www.gap-system.org`.

21. G.J Ellis. *GAP package HAP – Homological Algebra Programming*, 2017. URL `http://www.gap-system.org/Packages/hap.html`.

22. Max Neunhöffer. *GAP package IO – Bindings for low level C library I/O routines*, 2017. URL `http://www.gap-system.org/Packages/io.html`.

23. Dane Flannery and E.A. O'Brien. Computing 2-cocycles for central extensions and relative difference sets. 28:1939–1955, 2000. DOI:10.1080/00927870008826937.

24. Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism, II. *Journal of Symbolic Computation*, 60(0):94 – 112, 2014. DOI:10.1016/j.jsc.2013.09.003.

25. Marc Keilberg. Applications of quasitriangular structures for the doubles of purely non-abelian groups. arXiv:1708.06583.